

# '68'

**\$2.50** USA

Australia  
Singapore

A \$ 4.00  
S \$ 8.00  
Malaysia

New Zealand: NZ \$ 4.00  
Hong Kong: H \$ 20.00  
M \$ 8.00

## MICRO JOURNAL

**VOLUME II ISSUE 5 • Devoted to the 68XX User • May 1980**  
"Small Computers Doing Big Things."

SERVING THE 6800 USERS WORLDWIDE

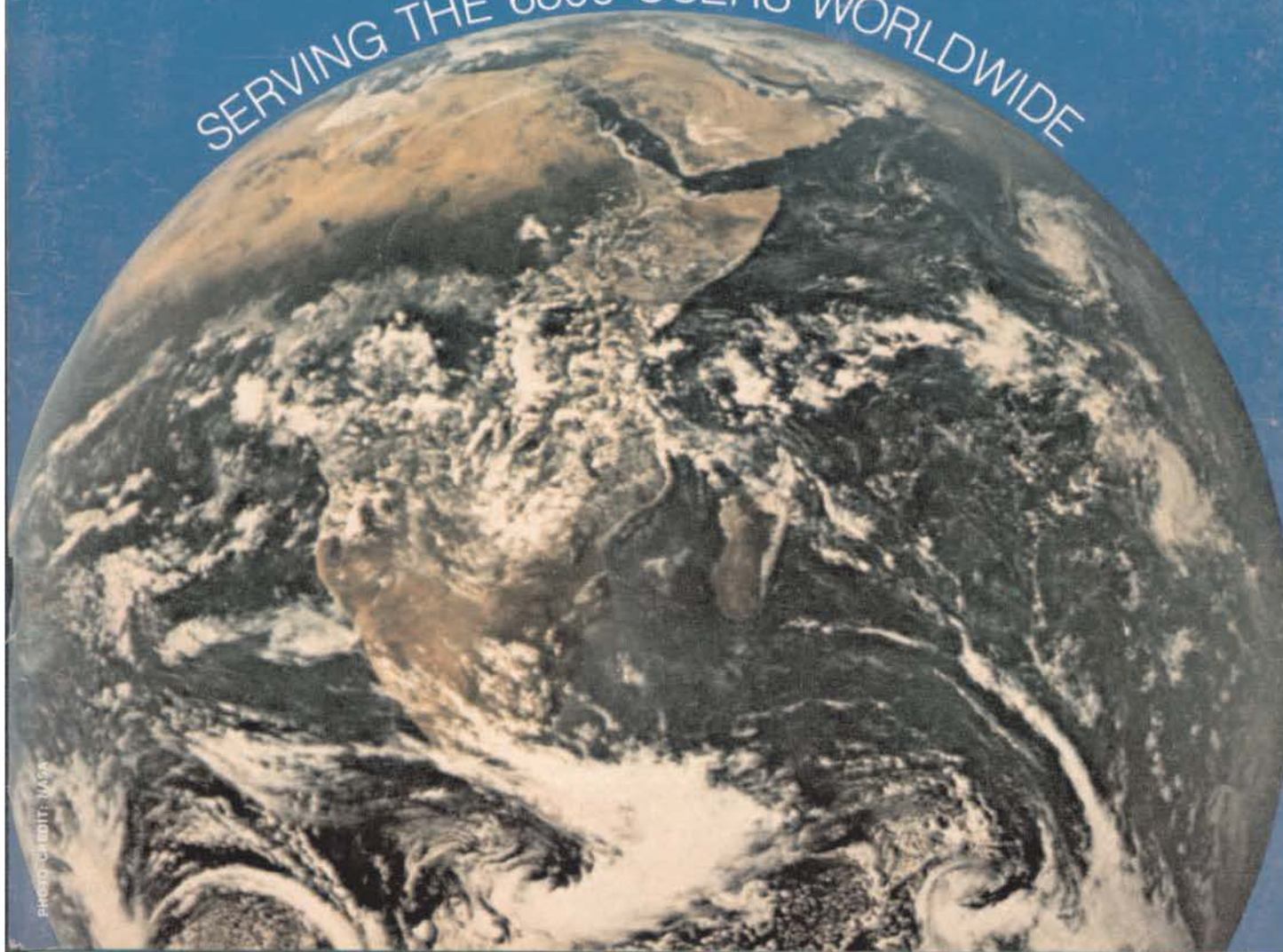


PHOTO CREDIT: NASA





## SYSTEMS - SOLUTIONS

If you have a problem that can be solved by a computer—we have a systems solution.

- Two central processors with maximum RAM capacities of 56K and 384 K bytes
- Three types of disk drives with capacities of 175K, 1.2M and 16M bytes
- Two dot matrix printers with 80 and 132 line capacity
- A Selectric typewriter interface and a daisy wheel printer

Match these to your exact need, add one or more of our intelligent terminals and put together a system from one source with guaranteed compatibility in both software and hardware.

Southwest Technical Products systems give you unmatched power, speed and versatility. They are packaged in custom designed woodgrain finished cabinets. Factory service and support on the entire system and local service is available in many cities.



SOUTHWEST TECHNICAL PRODUCTS CORPORATION  
219 W. RHAPSODY  
SAN ANTONIO, TEXAS 78216

(512) 344-0241



# The **FLEX**™ Disk Operating System

FLEX™ is a powerful, easy-to-use disk operating system which has become the standard for 6800 and 6809 micro-processor systems. Two new versions are now available and both include a disk editor and assembler:

## **FLEX for the EXORciser™ \$150.00**

Runs on a Motorola EXORciser with EXORdisk™ II or III. Requires no hardware modifications with the possible exception of memory re-addressing. Uses the same boot as MDOS™.

## **FLEX for General Use \$150.00**

Fully documented to allow a user to write his own terminal and disk I/O routines to adapt to most any hardware. Three system requirements are: (1) at least 12K of RAM at \$0000; (2) 8K of RAM at \$A000 for 6800 or \$C000 for 6809; (3) floppy disk drive capable of 256-byte, soft sectors. This package is not for beginners!

## **FLEX Support Software**

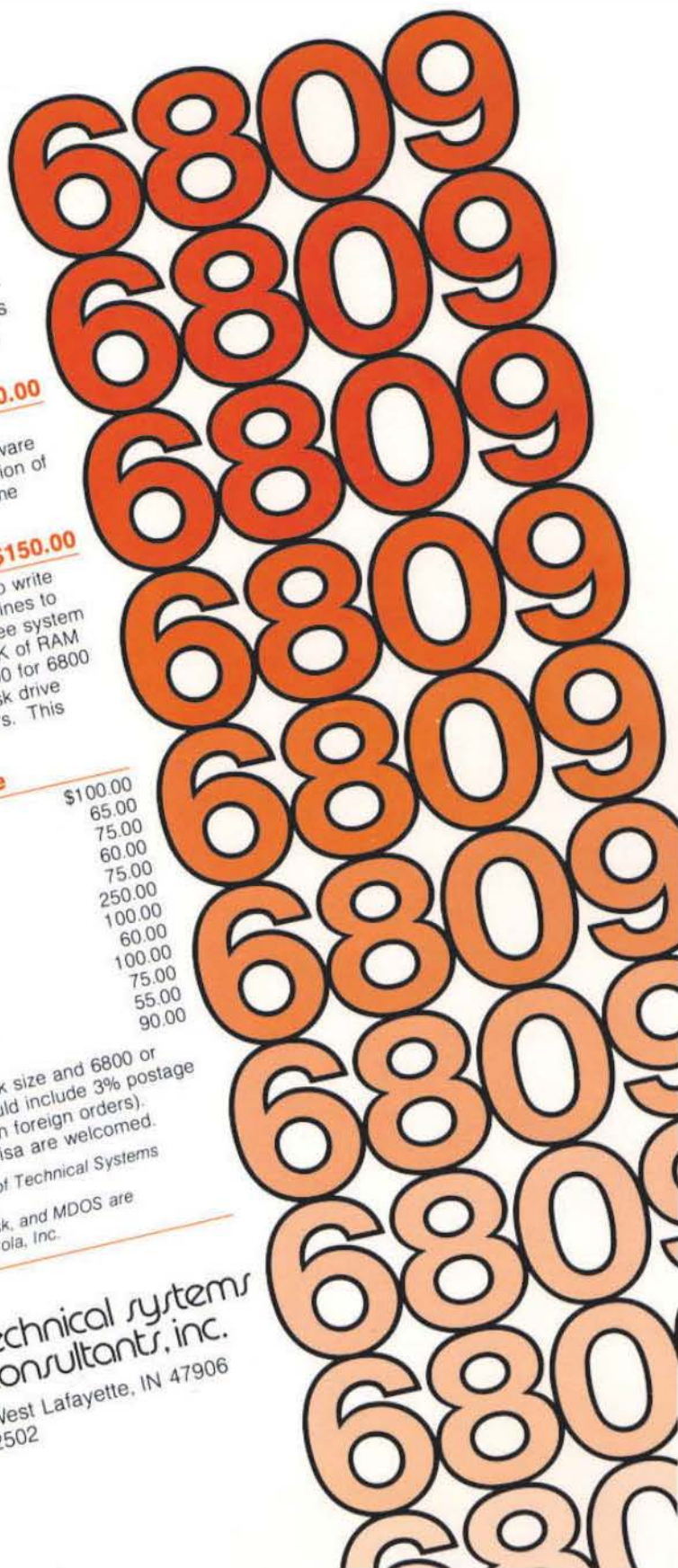
<b>New!</b> Extended BASIC	\$100.00
Standard BASIC	65.00
6809 Diagnostics Package	75.00
Text Processing System	60.00
Sort/Merge	75.00
68000 Cross Assembler	250.00
6809 Cross Assembler	100.00
6809 FLEX Utilities	60.00
6800 FLEX Utilities	100.00
6809 Debug Package	75.00
6800 Debug Package	55.00
FLEX for SWTPc	90.00

Be sure to specify disk size and 6800 or 6809. All orders should include 3% postage and handling (10% on foreign orders). Mastercharge and Visa are welcomed.

FLEX is a trademark of Technical Systems Consultants, Inc.  
EXORciser, EXORdisk, and MDOS are trademarks of Motorola, Inc.



**technical systems  
consultants, inc.**  
Box 2570, West Lafayette, IN 47906  
(317) 463-2502



# '68'

Portions of text prepared using the following.

SWTPC 6800-6809-DMAF2-CDS1-CT82-Sprint 3  
Southwest Technical Products  
219 W. Rhapsody  
San Antonio, Texas 78216

GIMIX Super Mainframe-Assorted memory boards  
GIMIX Inc.  
1337 West 37th Place  
Chicago, IL 60609

Publisher: Don Williams Sr.

Executive Editor: Larry Williams

Subscriptions and Office manager  
Mary Robertson

General Girl 'Friday'  
Joyce Williams

Contributing Editors:

Mickey Ferguson  
Dennis Womack  
Dr. Jack Bryant  
Dr. Chuck Adams  
Dr. Theo Elbert  
Dr. Jeffery Brownstein  
Dale Puckett  
Russell Gore  
T. Jackson, Japan

Typography and color work:  
Williams Inc.  
Chattanooga, TN 37421

## \* CONTENTS \*

SWTPC MP-09 CPU.....8	Clark
MPI51/52 DISK DRIVES...8	Pass
A HOBBYIST SPEAKS.....9	Libby
SSB DOS - Ver. 5.1.....10	Puckett
MINIDISK+ DOS.....11	Review
BASIC UTILITY PACKAGE..14	Puckett
STUFF FOR 6808.....16	Jones
BCD MULTIPLY.....19	Visher
MEK-D2 TO S50 BUS.....21	Phelps
CLASSIFIED ADS.....21	
HELP.....21	
BOOKKEEPING (final?)....28	Stock
BIT BUCKET.....30	

# MICRO JOURNAL

## Send All Correspondence To:

'68' Micro Journal  
3018 Hamill Rd.  
PO Box 849  
Hixson, Tennessee 37343

— Phone —  
Office: 615-870-1993  
Plant: 615-892-7544  
Copyright © 1980

'68' Micro Journal is published 12 times a year by '68' Micro Journal, 6131 Airways Blvd., Chattanooga, TN 37421. Second Class postage paid at Chattanooga, TN. Postmaster: Send Form 3579 to '68' Micro Journal, PO Box 849, Hixson, TN 37343.

1-Year \$14.50 2 Years \$26.00 3 Years \$36.50

## —ITEMS SUBMITTED FOR PUBLICATION—

(Letters to the Editor for Publication) All letters to the Editor should be substantiated by facts. Opinions should be indicated as such. All letters must be signed. We are interested in receiving letters that will benefit or alert our readers. Praise as well as gripes is always good subject matter. Your name may be withheld upon request. If you have had a good experience with a 6800 vendor please put it in a letter. If the experience was bad put that in a letter also. Remember, if you tell us who they are then it is only fair that your name not be withheld. This means that all letters published, of a critical nature, cannot have a name withheld. We will attempt to publish 'verbatim' letters that are composed using 'good taste.' We reserve the right to define (for '68' Micro) what constitutes 'good taste.'

(Articles and items submitted for publication) Please, always include your full name, address, and telephone number. Date and number all sheets. TYPE them if you can, poorly handwritten copy is sometimes the difference between go, no-go. All items should be on 8X11 inch, white paper. Most all art work will be reproduced photographically, this includes all listings, diagrams and other non-text material. All typewritten copy should be done with a NEW RIBBON. All hand drawn art should be black on white paper. Please no hand written code items over 50 bytes. Neatly typed copy will be directly reproduced. Column width should be 3 1/4 inches.

(Advertising) Any Classified: Maximum 20 words. All single letters and/or numbers will be considered one (1) word. No Commercial or Business Type Classified advertising. Classified ads will be published in our standard format. Classified ads \$7.50 one time run, paid in advance.

Commercial and/or Business advertisers please write or phone for current rate sheet and publication lag time.



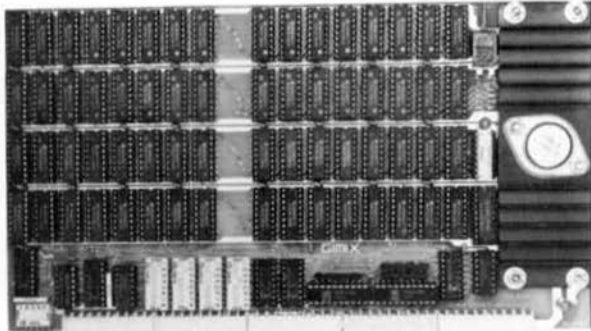


# WANT TO BE KING OF THE HILL?

TREAT YOURSELF ROYALLY WITH GIMIX UNIQUE AND INCOMPARABLE  
BOARDS AND SYSTEMS... DIP-switch Versatility for use with  
both SS50 (6800) and SS50C (6809) Systems (SWTP. etc.)

## 32K STATIC RAM BOARDS

Designed for use with: ★ Existing SS50 Systems  
★ SS50C Extended Address Systems



- Assembled
- Burned In
- Tested

16K . . . \$328.12

24K . . . \$438.14

32K . . . \$548.15

16K and 24K Versions are  
socketed for 32K and re-  
quire only additional  
2114's for expansion.

### FEATURES:

- Decoding for 4 Extended Address Lines (allows memory decoding up to 1 megabyte)
- DIP-switch to set extended addressing or disable it
- 4 separate 8K blocks, addressable to any 8K boundary by DIP-switch
- Each 8K block may be individually disabled
- Write protect either of two 16K sections
- Low power consumption — uses 2114L low power RAMS — (2 amps typical for 32K)
- Fully Socketed
- Gold Bus Connectors
- Guaranteed 2MHz operation

### FACTORY PRIME STATIC RAMS

2114L 450 ns . . . . .	\$5.90	200 ns . . . . .	\$6.90
4044 450 ns . . . . .	\$5.90	250 ns . . . . .	\$6.90

ADD \$5.00 HANDLING ON ORDERS UNDER \$200.00

## I.O. BOARDS NEW!! 50 PIN 8 PORT RS-232 SERIAL BOARD

- ★ With SS50C extended addressing (can be disabled)
  - ★ DIP-switch selectable Baud Rates for each Port
  - ★ Occupies 16 bytes of address space
  - ★ Dip switch addressable to any 16 byte boundary
  - ★ Any 2 of IRQ, NMI or FIRQ can be used
  - ★ RS-232 handshaking for each port (CTS, RTS, DCD)
- ..... \$288.40
- ★ With optional on Board Baud Rate Generator
- ..... \$318.46

### ALSO AVAILABLE

50 Pin 8 Port Parallel	\$198.45
30 Pin 2 Port Parallel	\$ 88.42
30 Pin 1 Port Serial (RS 232 or 20MA, current loop)	\$ 88.41
Cable Sets for above	\$ 18.95

### EXPORT NOTES:

For 50Hz 230V C.V. POWER SUPPLY . . . . . Add \$30.00  
80 x 24 VIDEO BOARDS — Specify Format (No Added Charge)

On Orders under \$250.00 for a Single Board, or Chips, please Add \$30.00 Handling and we will ship Air Mail Prepaid.

On all other orders we will ship via Emery Air Freight Collect, and we will charge no handling. All orders must be prepaid in U.S. Funds. Please note that foreign checks have been taking about eight weeks for collection, so we would advise wiring money or checks drawn on a bank account in the U.S. Our bank is the Continental Illinois National Bank of Chicago, account #73-32033. Visa or Master Charge also accepted.

GIMIX® and GHOST® are Registered Trademarks of GIMIX INC.

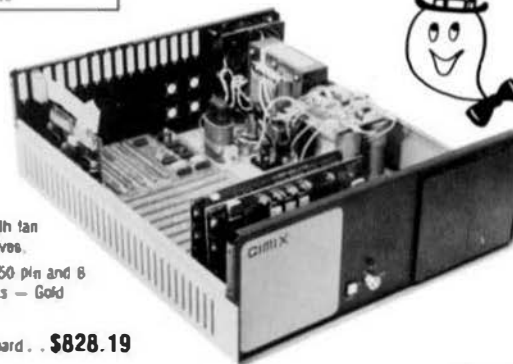
## THE CLASSY CHASSIS

- Ferro-resonant Power Supply
- Heavy weight Aluminum Cabinet with fan and provisions for two 5" disk drives.
- 6800/6809 Mother Board, fifteen 50 pin and 8 DIP-switch addressable 30 pin slots — Gold Plated Pins, Fully Decoded.
- Baud Rate Generator on Mother Board . . \$828.19

**32K SYSTEM** *Incomparable Features at a Comparable Price!* . . . . . \$1,594.59

Includes: Chassis, 6800 CPU, 32K RAM Board, Choice of I/O Card.

**16K Version of above** . . . . . \$1,374.49



Phone, write, or see your dealer for details and prices on our broad range of Boards and Systems for the SS50/SS50C bus and our AC Power Control Products for all computers.



# GIMIX inc.

*The Company that delivers  
Quality Electronic products since 1975.*

1137 WEST 37th PLACE, CHICAGO, IL 60609  
(312) 927-5510 • TWX 910-221-4055

SEE GHOST AD PAGE 44



## FUTURE FEATURES on the FIFTY

The **FIFTY BUS** gives you compatibility and choices of Hardware and Software offerings by other manufacturers and Software houses such as:

- SWTP
- TSC
- Microware
- Microworks

And now **GIMIX** presents our **SS50C 6809 CPU** card and systems.

The 6809 CPU card will be available in a standard version and our 6809 PLUS version that is fully socketed to allow adding options at anytime.

- + A 6840 timer package that provides 3 independent 16 bit counters is included on all 6809 PLUS cards.
- + A 9511 or 9512 Arithmetic Processors option with its own independent crystal that allows you to use 2, 3, or 4 MHz parts in any combination with the 6809 running at 1, 1.5, or 2 MHz.
- + 1K of scratchpad RAM
- + A Time of Day Clock option with battery back-up. With this option you can also substitute 1K of CMOS RAM that will also be battery backed up.
- + User selectable processor speeds without having to change the crystal.

32K of PROM, ROM or RAM. Both versions have 4 sockets that can each hold from 1K to 8K parts. Single or multiple voltage parts can be used on the PLUS version. The standard version only allows the use of single voltage parts.

All on board devices and options can use extended addressing so that they will only respond to that page to which they are set.

The card is double buffered and allows versatility in the use of software and memory address control disciplines.

Please note that this card does not have an on board baud rate generator, and must be used in systems where baud rates (if needed) are provided elsewhere in the system.



And looking into our Crystal Ball we are hoping to ship by the 3rd Quarter of 1980 our:

### GIMIX DISC CONTROLLER CARD

Like all **GIMIX** products, it is designed for reliability. The board uses a phase lock loop data separator. It will use DMA and can control up to four 5" or four 8" single or double sided, single or double density drives.

We plan to have the latest generation of software available including: TSC's FLEX and forthcoming UNIFLEX, Microware's/Motorola's OS-9 and BASIC 09, and CSI's UCSD Pascal.

By the 2nd Quarter of 1980 we should be delivering the

### GIMIX HIGH RESOLUTION GRAPHICS CARD SET

It is capable of 512 x 512 resolution using 32K of Static RAM. It needs 8K of memory space on the bus, can use extended addressing, and can occupy 32K, if desired.

GHOSTable. On board software control registers allow the memory to disappear and reappear on the bus.

For Color, 3 of these sets can be used with RGB monitors.

Supplied with fast, compact and powerful software driver routines including an interface for TSC's BASIC. The software supports both vector and character operations, with control of virtually all board features.

It will also be available in 256 x 256 and 640 x 240 resolution versions.



And looking further into the Future...the 68000 should arrive in 1981

a **GIMIX 68000 CPU** card that will not be a whole new ball game... just a new CPU card that is being designed for use with our present mainframe that has a 15-50 pin and 8-30 pin Motherboard, 25 amp power supply, Memory and other I.O. and video cards, and our forthcoming Disc Controller Card.

**GIMIX** inc.

1337 WEST 37th PLACE • CHICAGO, IL 60609 • (312) 927-5510 • TWX 910-221-4055

GIMIX® and GHOST® are registered trademarks of GIMIX Inc.

© 1980 GIMIX Inc.



# 5-1/4" Minidisk — Soft or Hard Sector

Dealer and Volume Discounts Available

S  
A  
V  
E

D  
I  
S  
K



**Verbatim**

MD 525-01 MIN.-10  
minidisk **\$3.09** ea.  
Disk No. Information Terminals

**DISK**

minidisk™

5" Soft Sector \$3.09 each  
5" 10-16 Sector \$3.09 each  
8" Single Side, Double  
Density \$3.75 each  
8" Double Side, Double  
Density \$5.75 each  
Minimum Order 10(1 box)  
Add \$3.00 for 5" Plastic Box  
Add \$4.00 for 8" Plastic Box

**Verbatim**

## SOUTH EAST MEDIA SUPPLY

P.O. Box 794

615-870-1993

Hixson, TN 37343

# In the world of 6800 Microcomputing there is only one Universal Mini-Disk System ...

## the PERCOM LFD-400™ with SOFTRAN™

Made possible by SOFTRAN™, an innovative \$24.95 translator program, the reliable Percom LFD-400™ has just been upgraded to the first universal mini-disk storage system.

Suddenly the two worlds of 6800 minidiskette software become one. Because the LFD-400™ with SOFTRAN™ can read either soft-sectored or hard-sectored disks.

And owning an LFD-400/SOFTRAN system means you can run minidiskette programs from the enormous combined selection of all of the principal 6800 software houses — TSC, Computerware, the Software Works, Hemenway Associates and of course Percom.

Available in versions for mini FLEX†, FLEX 2.0† and Smoke Signal Broadcasting Company's DOS, SOFTRAN™ copies soft-sectored minidiskettes track-for-track onto hard-sectored minidiskettes. If the source disk includes a FLEX† or 'Smoke' DOS, SOFTRAN™ is used to modify the operating system to function with the Percom LFD-400™.

SOFTRAN™ is supplied on a minidiskette along with utilities for only \$24.95. A users manual is included. You must indicate whether SOFTRAN™ is to be used for mini FLEX†, FLEX 2.0† or Smoke's DOS.

The Percom LFD-400™ mini-disk system sells for

only \$599.95, complete with: (1) the drive, drive electronics and Percom's rugged PS-401 power supply all in a finished enclosure, (2) a demonstrably superior controller PC card featuring an explicit data/ clock separation circuit, MPX, a remarkable 2K DOS, and provision for 1K extra PROM, (3) an interconnecting cable and (4) a 70-page users manual.

Also available: Upgrade kits for SWTP or 'Smoke' mini-disk drive systems. Kit includes LFD-400™ controller, MPX DOS & SOFTRAN™. Only \$224.95.

Available soon!

SOFTRAN™ for Percom's 77-track LFD-800™ mini-disk system; SOFTRAN/9™ for 6809 FLEX† files and programs.



™ trademark of Percom Data Company, Inc.

† trademark of Technical Systems Consultants, Inc.

# PERCOM

PERCOM DATA COMPANY, INC.  
211 N. KIRBY GARLAND, TEXAS 75042  
(214) 272-3421

Percom 'peripherals for personal computing'

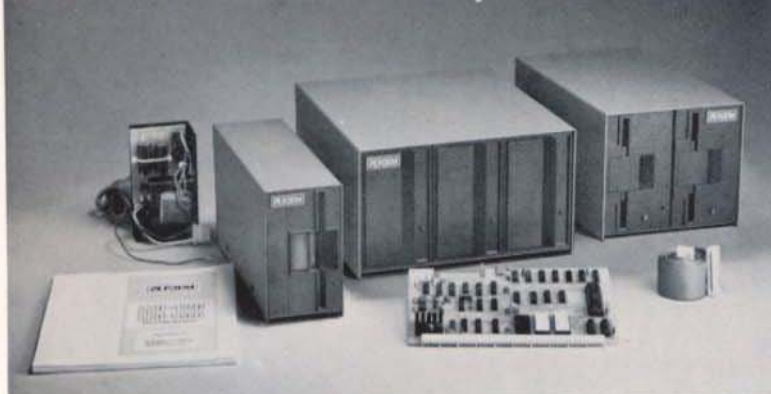
To place an order or request additional literature call toll-free 1-800-527-1592. For technical information call (214) 272-3421. Orders may be paid by check, money order, COD or charged to a VISA or Master Charge account. Texas residents must add 5% sales tax.

PRICES AND SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE



# A Few Extraordinary Products for Your 6800/6809 Computer

## SS-50 Bus LFD-400™ and LFD-800™ Systems



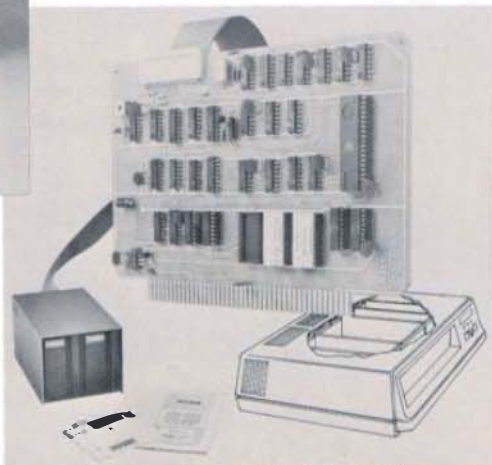
Percom mini-disk systems start as low as \$599.95, ready to plug in and run. You can't get better quality or a broader selection of disk software from any other microcomputer disk system manufacturer — at any price!

**Features:** 1-, 2- and 3-drive systems in 40- and 77-track versions store 102K- to 591K-bytes of random access data on-line • controllers include explicit clock/data separation circuit, motor inactivity time-out cir-

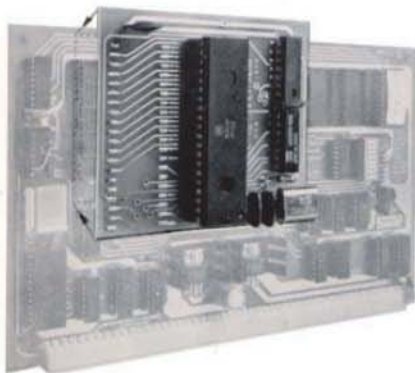
cuit, buffered control lines and other mature design concepts • ROM DOS included with SS-50 bus version — optional DOSs for EXORciser\* bus • extra PROM sockets on-board • EXORciser\* bus version has 1K-byte RAM • supported by extended disk operating systems; assemblers and other program development/debugging aids: BASIC, FORTRAN, Pascal and SPL/M languages; and, business application programs.

From Percom . . .

**Low Cost  
Mini-Disk Storage  
in the Size You Want**



EXORciser\* Bus LFD-400EX™ -800EX™ Systems



### Upgrade to 6809 Computing Power

This 6809 upgrade adapter may be used on the SWTP 6800 and most other 6800/6802 MPU cards. Supplied assembled and tested, it costs only \$69.95 with user instructions. The original system may be restored by merely unplugging the adapter and a wire-jumpered DIP header, and re-inserting the original components. Also available for your upgrade computer is PSYMON™, the Percom SYstem MONitor for the Percom 6809 single-board computer. PSYMON™ on 2716 ROM costs only \$69.95 — PSYMON™ is also available on minidiskette, with source and object files, from the Percom Users Group.

™ trademark of Percom Data Company, Inc.  
• trademark of the Motorola Corporation.

Prices and specifications subject to change without notice.

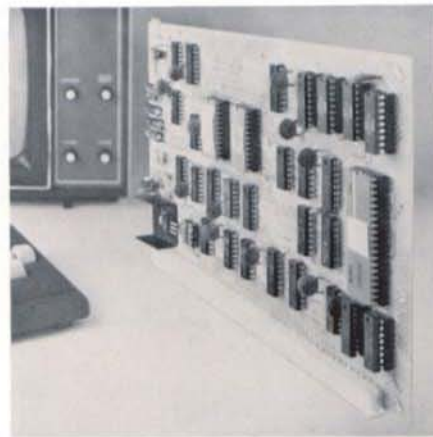
### Full Feature Prototyping PC Boards

Percom SS-50 and I/O bus prototyping cards include all of the features needed for easy, straightforward prototyping. Use wire wrap, wiring pencil or solder wiring. **Features:** tin-lead plating over 2 oz. copper wets quickly, solders easily • provision for power regulators and distributed capacitor bypassing • SS-50 bus card accommodates 34- and

50-pin ribbon connectors on top edge, 10-pin Molex connector on side edge — costs only \$24.95. • I/O bus card is 1-1/4" higher than SWTP I/O card, accommodates 34-pin ribbon connector and 12-pin Molex connector on top edge — costs only \$14.95. • Both card designs accept 14-, 16-, 24- and 40-pin DIP sockets.

### The Electric Window™: Instant, Real-Time Video Display Control

This VDC card resides completely in main memory so that control is accomplished instantly by direct MPU access to the on-board 2K character-store memory and the display control registers. Price is only \$249.95. **Features:** Programmable CRT controller chip provides extraordinary capability for software control of functions such as number of characters per line, number of lines displayed, highlighting and interlaced or non-interlaced scan • includes ASCII 128-unit character generator which generates 7-dot by 12-dot characters — lower case letters have descenders • provision for optional ROM for special characters/symbols • comprehensive manual includes full listing of WINDEX™, the Electric Window™ driver program — WINDEX™ is also available on minidiskette through the Percom Users Group.



**PERCOM**

PERCOM DATA COMPANY, INC.  
211 N. KIRBY GARLAND, TEXAS 75042  
(214) 272-3421

Products are available at Percom dealers nationwide. Call toll-free, 1-800-527-1592, for the address of your nearest dealer, or to order direct.

## THE SWTPC MP-09 CPU CARD

A few months ago I purchased a new CPU board for my system to upgrade it for operation with the 6809 uProcessor. I have been pleased with the MP-09 CPU board from Southwest Technical Products and was therefore somewhat surprised at the comments in reference to it in the January 1980 issue of KiloBaud Microcomputing. Perhaps the author's trepidation over the amount of effort required to physically accomplish the conversion and "horror" of having memory respond at an address other than where it is strapped is really only a result of lack of familiarity with the new approach. If that is the case than some explanation ought to alter his opinion.

I am operating my board in a Midwest Scientific Instruments mainframe and the physical switch was almost trivial. The only change required to the mainframe was an addition of a two wire reset cable (connector supplied) and a flick of a dip switch to move I/O to \$E000. Furthermore to change back to 6800 requires only that I plug the reset cable back onto the motherboard and dip select I/O at \$8000. The one little problem I did have was a result of differences between the SWTPC mainframe and the MSI mainframe. In the SWTPC system the FIRQ, IRQ, and HALT lines are pulled up to +5 volts on the motherboard; whereas, they are not in the MSI system. The solution was to simply add pullup resistors to the MP-09 CPU board.

As far as the "dynamic address translation" system is concerned I find it actually makes it easier when switching between FLEX2.0 and FLEX09. The reason for this is that FLEX2.0 requires RAM at \$A000-\$BFFF so that is where I dip select my memory, but when I convert to FLEX09 it needs RAM at \$C000-\$DFFF and I only have RAM at \$D000. Therefore, the "dynamic address translator" takes the 4K block of RAM at \$B000 and relocates to \$C000 and moves the 4K block at \$A000 and moves it down as far as required to give me contiguous RAM memory from \$0000 up to end of user RAM. This relieves me of the burden of having to reconfigure memory when converting between 6800 and 6809.

However, the issue was raised as to what effect this would have on others attempting to write compatible software for a system whose memory wandered around. In point of fact memory is relocated in a very orderly fashion and if the programmer so desires he can move it where he chooses. Perhaps an explanation of the DAT system is in order.

During power-up initialization the processor initializes the DAT RAM chip (IC11 of the MP-09 board) to contain the following....  
\$0F \$0E \$0D \$0C \$0B \$0A \$09 \$08 \$07 \$06 \$05 \$04  
\$03 \$02 \$01 \$00

....to cause all physical addresses to be equal to logical addresses. It then loads a test pattern of \$55AA to \$DOAO, \$COAO, \$BOAO, \$AOAO, \$90AO, \$80AO.....\$00AO and tests each location to see if the pattern wrote correctly to that location. The first location that it finds that it can write to it chooses as the 4K block of RAM that it will relocate to \$D000. The relocation is accomplished by inverting the most significant four bits of the address of the first 4K block located and then writing that 4 bit pattern into the DAT RAM chip at location \$FFFD. This causes the physical address of that 4K block of RAM to be translated so it logically responds at as address of \$0000. Now that the system knows there is RAM at \$D000 it initializes the stack pointer to \$DFC0 and creates a logical to real address (LRA) table from \$DFD0 to

\$DFFF. The table is created by several operations. One clears \$DFD0 to \$DFDC. Another presets \$DFDE and \$DFDF to \$F1 and \$F0 respectively. The third operation uses a test pattern method of writing \$55AA to the remaining 4K blocks at location \$X0A0 not previously tested for RAM to locate the remaining RAM in the user system. Once located the inverse of the most significant 4 bits of the physical address are saved at their corresponding location in the LRA table. Table locations corresponding to 4K blocks where no RAM is located are left cleared. For example in a system that had RAM physically addressed at the following locations.....

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
4K	4K	4K	4K	4K	4K	4K	4K	---	4K	4K	4K	4K	---	---	---

.....the table created from \$DFD0 to \$DFFF would be.....

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0F	0E	0D	0C	0B	0A	09	08	00	06	05	04	00	03	F1	F0

Now the system tests location \$DFDC. If \$DFDC contains a zero value indicating that no RAM is physically addressed at \$C000 which FLEX09 requires; then sequentially lower locations in the table are tested until the first non-zero value is located. This non-zero value is transferred to \$DFDC and cleared from its existing location in the table thus translating that block of memory to logically respond at \$C000. Next the table is compressed toward \$DFD0 to eliminate any zero locations in preparation for ensuring that all remaining RAM will be logically addressed in a contiguous block from \$0000 to the end of user RAM. Finally the table created that now is configured as follows for the previous example.....

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0F	0E	0D	0C	0B	0A	09	08	06	05	00	00	04	03	F1	F0

the address translation. The user system will now appear to have RAM logically addressed as follows.....

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
4K	4K	4K	4K	4K	4K	4K	4K	4K	4K	---	---	4K	4K	---	---

Once the user understands the DAT system it is a trivial matter to use it in conjunction with his custom firmware. The user has the option of writing his own relocation pattern in the DAT chip or he may fool the system by placing a test pattern of \$55AA in location \$X0A0 of his ROM.

By: Allen Clark  
2502 Regal Oaks Lane  
Lutz, Fla. 33549

This article will be followed in the next few months with another series of articles, by the same author, to include a complete commented source listing of the SWTPC SBUG-E monitor and a detailed discussion on the 'DAT' portion of this popular 6809 monitor.

DMW

-----

Dr. Bud Pass  
1454 Latta Lane NW  
Conyers, GA 30207

## THE MPI51/52 DISK DRIVES

The MPI51 and MPI52 are Shugart-SA-400-compatible 5 1/4" Mini-Floppy disk drives. Both are capable of double-density operation and the MPI52 is capable of double-sided operation. The MPI51 provides 40 tracks, as opposed to 35 on the SA400. Track-to-



track access time on the MPI drives is five milliseconds, whereas it is 40 milliseconds on the SA400. The MPI drives are often available discounted and may be substantially less expensive and more available than the SA400.

Since the MPI drives are compatible with the SA400, the SWTPC DC-2 Mini-Floppy Disk Controller board (or similar boards such as the PTA FD-1) may be used to interface them to a 68XX SS-10 bus. This will allow FLEX and other operating systems to be used without modification, as long as certain restrictions are observed. Since the 1771 does not support double-density encoding, single-density must be used. Double-sided operation and the use of tracks 36-40 are also not supported by the unmodified FLEX software and DC-2 hardware. FLEX will also not generally take full advantage of the faster stepping speed. Without changing FLEX and the DC-2 (or FD-1) controller board, only four logical drive numbers are available.

Patches to allow the use of 40 tracks in FLEX have been printed previously in '68 MICRO. The only changes required for FLEX 2 and FLEX 9 are in the NEWDISK module. Patches to speed up head movement are as follows: change 1B to 18 at locations BEFF and BF5F in FLEX2 or locations DE96 and DEFA in FLEX9.

There are two lines of special interest here in the MPI drive interface cable which are not in the Shugart drive interface cable. They are drive-select-4 (line 6) and side-select (line 32). As with all the other lines in the cable, they are logically active-low.

The following table provides connection information for drive-select-4:

	Cut	Add
DC-2	IC4-8	IC4-8 to J1-6
FD-1	no	7442-4 to F1-6

This connection is necessary only if four drives are to be attached to the system. It may be performed, however, even with fewer than four drives attached.

There are several means of implementing side-select on the MPI52 drives. None should be used for MPI51 drives. Since side-select on the MPI52 works differently from the double-sided WANGCO drives, FLEX 9 will not support double-sided operations directly, as will not FLEX 1 and FLEX 2. However, by limiting the number of drives to two and treating each side as a logical drive number, double-sided operation may be accomplished with no software changes and only minor hardware changes. In this scheme below, the fronts of the drives are numbered 0 and 1 and the backs are numbered 2 and 3. The following table provides connection information for side-select according to this method:

	CUT	ADD
DC-2	IC5-3, IC4-5, IC4-6	IC5-3 to IC5-8, IC8-15 to IC4-5 1K PULLUP J1-32 to IC5-16 IC4-5 to J1-32
FD-1	7442-14	74175-6 to F1-32

Software and hardware changes would be necessary to increase the number of logical drives beyond 4 and the number of physical double-sided MPI52 drives beyond 2.

One other hardware change may be made to the disk controller board, regardless of the type of mini-floppy being used (even non-MPI). This will allow the use of a manual SPDT switch on the controller board to logically swap drives. The switch may be installed to even the use of drives, or to help get a system up when drive 0 does not want to read the boottrap program. The following table provides connection information for the switch:

	CUT	ADD
DC-2	IC5-2	IC5-2 to wiper, IC8-10 to throw-1, IC8-11 to throw-2
FD-1	7742-15	7442-15 to wiper, 74175-3 to throw-1, 74175-2 to throw-2

The MPI drives are quite reliable, available, and relatively inexpensive. They are even MORE useful with the modifications described above.

## A HOBBYIST SPEAKS OUT

All people interested in computer systems for any reasons whatever make \$1,000 a week, are senior systems analysts and have no other compunctions in life than to serve computer and vendor alike - or so it would seem if one is to believe the ads and articles that make up the majority of the microprocessor oriented "hobbyist" periodicals.

During a moment of reflection after thumbing through the latest issues of BYTE and '68' Micro Journal it occurred to me that surely not everyone was going out and buying the latest muscletron systems with multifloppies or hard disks, 128k and software packages that attempt to emulate the biggies in power, if not speed or pain. But what ever happened to the "little" guy. You remember, the one with a few bucks that just wants to have some creative fun and is content with a smallish, smokeable tape based system that meets his idea of adventure? I'm convinced he is still around someplace, even if not heard from that often. And how is someone going to be interested in entering this hobby if he is faced with a minimum \$1500 outlay to just get started?

Therefore, I propose a small system 68xx user group be formed, or at least a loose confederacy of users that are not yet lost in the bigness craze that has so taken the microprocessor world of late. An easy way to start, I believe, is for the people with tape systems, or just even scratch built smokers, to write in so a list can be published of such people and their systems. This would make for much better communication and information

interchange. We might even be able to come up with some good ideas.

If you have any comments about this write in to *68 Micro Journal* or to me, I'd like to know how many tape users are still in existence. Sticking with our "old" systems may not be so bad after all. For while others are rushing headlong into floppies, hard disks, and the *6800*, we could be saving our pennies for the advent of the true 16 bit dream machines coming down the pike.

Mark Libby  
3923 Lynncrest Dr  
Cleveland, TN. 37311  
(My system consists of:  
20k SWTPC 6800  
Percom CIS-30+ and JPC tape  
JPC/CFM/3 tapmon.  
CT-64 and Anderson J.  
terminals.)

#### SSB DOS - VERSION 5.1

Dale Puckett  
14753 Endsley  
Woodbridge, VA 22193

This review looks at some of the new features of DOS68, version 5.1, for the Smoke Signal Broadcasting BFD-68, LFD-68 and Chieftain disk systems. The upgraded disk operating system is available from Smoke Signal Broadcasting, 31336 Via Collinas, Westlake Village, CA, 91361 and from Computerware, 1512 Encinitas Blvd., Box 668, Encinitas, CA, 92024. Computerware's price is \$75 according to a recent flyer. This price includes a new Editor which provides hardcopy output and automatic file back-up, plus, a new assembler which has multiple file input capabilities, line numbering and built in hardcopy. Since this software is an upgrade this review will compare this version with earlier releases in an effort to give *68 Micro* readers information they may want to consider if they are trying to decide whether or not to move up. When I first purchased my SSB system it was the only one available with a DOS (version 2.7) that worked. I learned to use it but some of its unhuman characteristics often upset me. Impersonal messages coupled with the system's inability to deal with any control characters in the input from the terminal were very frustrating. My terminal used certain control characters to backspace, home the cursor and clear the screen, etc., and the original DOS didn't even allow me to clear the screen without getting a nasty error message. All those problems are gone in version 5.1. In fact, the DOS is even supplied with a utility command that allows you to set a number of terminal parameters. We'll look at the following new commands which have been added: BACKUP, BUILD, EXEC, GOTO, P, REPAIR, SAVET, AND SET. DOS68 has two parts. The first is a monitor called TMON, the second, a disk file management system, DFM68. Both may be used by the programmer and all the necessary documentation is supplied. One feature provided by Smoke Signal with this version is a big time saver.

On the system disk you receive a file called SYSEQU.TXT which is designed to be included in the file list when assembling your programs. It contains all the equates necessary to interface with TMON and DFM68. It sets up the monitor entry points, the parameter table, all DFM linkages, etc. This feature was not provided when I bought my system or when I upgraded to version 4.1 last year. It is an excellent addition and when listed on a printer makes a handy reference for programmers. Additional monitor entry points that were not available with earlier versions, or at least not advertised, include: ZADDX, add the B-register to the index register; ZOUTHX, print a hex byte; ZOUTHM, print a hex address; ZPEEK, looks at next character in line buffer; ZOUTCH, an output vector users can change; ZPUTCH, a directed output vector; ZSTAT, checks the terminal input status; ZRESTR, restores I/O vectors; DCMOLN, calls DO processor; ZEXCMD, executes a command; ZLOAD, a file loader; ZNAMEJ, decodes a name ZORLF, prints a carriage return and line feed; and ZSTEXT, enters a default file extension. All of these routines are pointed to by a jump table in the beginning of the monitor. This makes them easier to use. One comment here for people who have been using MIKBUG or similar monitors. The routine ZOUTST, outputs a string pointed to by the X-register just like PDATA! in MIKBUG. Smoke Signal, however, uses a null or \$00 hex as a terminator instead of the usual 04. I really wish they had made it standard with the rest of the 6800 world. The routine ZSTEXT should also be mentioned as it was unavailable in earlier versions of DOS68. It allows the programmer to set a default extension by placing a numeric code in the B-register. Default extensions available include: 1, BIN; 2, TXT; 3, SRC; 4, CTL; 5, BAK; 6, OAT; 7, FOR; 8, TMP; and 9, a spare. All can be redefined by using the SET command. On the disk file management side there are several new function codes that were not included in earlier versions. These include: QFREE, report amount of free space available; QAPP, append two files; QRAFC, read the active FCB chain; QLOGD, logs a system drive so DFM can find its overlays; QCRF, create a random file; QORF, open a random file; and QERF, expand a random file. The latter function is very handy as it allows a programmer to make his files dynamic. They can grow if more space is required. The same function allows the user to initialize every byte in the newly expanded part of the file to any value he chooses.

#### Print Drivers

The older versions of DOS68 were a pain to use with line printers that required special software. You had to find the output vector in every place of software you used and then figure out a way to change it to point to your own routines every time you wanted to use it. Version 5.1 solves that problem by providing a "PP" command. Parallel, PPRINT.SYS and Serial, SPRINT.SYS drivers are supplied and are ready to run on most printers. A typical command line would read: P LIST 1. This command would print a directory of the disk in drive number 1 on your line printer. LIST 1, will send it to the terminal. Printer routines are loaded and initialized by giving the command: RUN PPRINT.SYS. Once this is done they are ready for use by the "PP" command. This is usually done when you boot up your system by placing the command in your STARTUP file.

#### COMMANDS

All commands are given to DOS68 one line at a time. Typing errors can be removed by typing control H, to backspace or control X, to delete the entire line. The difference is that these are the default values and they may be changed by using the SET command. Commands may also be repeated. This is done by typing control D immediately after the original execution. You may also recall your last command by



typing a control A. This will cause DOS to display the previously executed command, leaving the line buffer pointer at the termination character. At this point you may edit the line by backspacing and then type a carriage return to execute it in its new form.

#### A Few Complaints

A few complaints are in order here. Filenames may only be six characters long. This is unhandy. Also if you desire to choose a specific drive you must type the file spec with a colon like: I:LIST. The colon is in a very unhandy position on a lot of keyboards and a period would be a lot handier. In the LIST command I would like to have the ability to LIST only the files with a certain extension, or maybe even files that start with a certain letter, or two letters. Version 2.7 allowed you to list either the command files or the rest so maybe we are taking a step backward in this area. APPEND gives you the capability to append one file to another. The syntax is: APPEND, THIS, THAT. When you use this command THIS is gone forever. I don't like this implementation and would much prefer something like: APPEND, THIS, THAT, THESE. THIS and THAT should both be intact after the APPEND, I feel. Let's look at the new version 5.1 commands. GOTO is very useful as it allow you to jump to a program that has already been loaded into memory. The target address can be represented by a one to four digit hex number. BACKUP allows the user to make a complete image copy of a disk. Data is transferred on a track by track basis. It is faster than COPY but there is a disadvantage. If the original disk has files which are physically scattered, the new disk will be the same and your access times won't be as good. If you want to insure that all files on the disk are contiguous, use COPY. When I typed this command, I received a "ROM ERROR MESSAGE." This message is given if the user has one of the earlier ROMs in his system. This is a blow to compatibility and I would prefer a command which would work with all firmware. The experience makes you feel you have been had. BUILD is a nice addition which allows you to type in a small command file quickly. The files it creates default to a CTL extension and are used by the EXEC command. The EXEC command is also a very worthwhile addition. It allows the user to process a text file as a list of commands. When it is running DOS thinks it is getting its input from the keyboard. In fact, when booting the system automatically runs the command, EXEC, STARTUP. This configures your terminals and sets any of the operating parameters of your system. FORMAT is a prompting command that allows you to initialize diskettes. After formatting a disk, it retains control and thereby allows you to format a number of diskettes at a time. It also automatically copies the DFM68 overlay files onto the new disk. Its output is cute, especially the way it reports the track numbers as it works. I do have one small complaint. At one point it prompts you with "HONEST?" I typed "Y" for yes a half dozen times before I tried typing "H" for honest. Really now. REPAIR allows you to recover files that have been accidentally deleted, etc. I had to buy my original copy on a Users Group disk so it is nice of Smoke Signal to include it with the basic operating system. SAVE is not a new command, but this version has a new feature that is quite useful. You may save multiple regions of memory by simply repeating the starting address, ending address pairs as many times as needed in the command line. This is a lot easier than appending a number of small files together. It also saves disk space. SAVET, by the way, is exactly the same as SAVE except it loads at \$0100 instead of in the Transient Command Area.

#### The SET Command

The SET command is the one that will most likely

convince you to upgrade. It allows you to set a number of system and terminal parameters, including: back space character, delete character, depth count, width count, the number of nulls output with each carriage return, the number of eject lines to be sent at the bottom of a page, the STOP or escape character, the CONT character, and the BREAK character. It also lets you toggle a WAIT flag that will allow you to pause at the end of a page if you need to change paper, etc. SET also allows you to clear the User Command Table without rebooting, define a Date string, and set a location called MEMMAX, and a lock which will allow you force all lowercase letters to uppercase if you have a terminal that doesn't understand lowercase letters, like a Model 15 teletype or unmodified CT-1024. MEMAX can be read by other programs and allows the programmer to define the upper limit for user programs. With SET you can also change the CRT control port address, the Hard-copy port address, and the monitor ROM's echo control byte. It also allows you to define the system and work drives. Commands default to the system drive, and target files default to the work drive. One feature would make SET handier. It should allow the user to look at the parameters, if he desires, before he SETs them. An example would be to type SET(cr) if you wanted to see how you have things configured and use the present format if you want to change something.

#### Conclusion

DOS68 version 5.1 is a dramatic step in the right direction. Its cursor control is impressive and makes your CT-1024 look like it's been to school. Error messages are a lot nicer now and many of the shortcomings of earlier versions have been eliminated. And, the fact that it is driven by a parameter table makes it very easy to customize.

#### MINIDISK+ DOS

A 68 Micro Journal Lab Review

MINIDISK+ is a ROM-based disk operating system for the Percom LFD-400 floppy disk system. It is written by Bill Vergona of Car-Comp Microcomputers, Las Vegas, who also wrote an early version of Percom's MINIDOS-PLUSX DOS (TM of Percom Data Co.) (Since MINIDISK+ is only useful if you have a Percom disk system, the rest of this review will assume some familiarity with Percom's hardware and software.)

MINIDISK+ is a 2K disk operating system supplied in two 2708 EPROMs. It plugs into the LFD-400 disk controller, next to the MINIDOS ROM already there. It adds named files to the basic read-write capability of MINIDOS itself. MINIDISK+ works with MINIDOS, but cannot be used with MINIDOS-PLUSX; the latter must be removed before plugging in the MINIDISK+ EPROMs, and the named file and directory handling functions of MINIDOS-PLUSX are then taken over by MINIDISK+.

Since MINIDOS itself stays, any program which uses only Percom's DSSS format ignores MINIDISK+ and works just as it did before it was installed. But programs written for MINIDOS-PLUSX do not work with MINIDISK+. Actually, this is not much of a problem since there is only one Percom program which fits into this category - Super Basic. Car-Comp solves this problem by providing patches to adapt Super Basic to work with the new DOS.

On the other hand, since most Percom software does not use named files, for anyone who primarily uses this software (which includes Percom's assembler, editor, text processor, or the Softran conversion which runs TSC's Flex or SS8's DOS68) changing from MINIDOS-PLUSX to MINIDISK+ would have no effect one way or the other.

When it comes to named files, MINIDISK+ is quite different from MINIDOS-PLUSX, and has a number of very interesting features, made possible by the fact that it occupies 2K instead of just 1K. Most of these make use of the system a bit easier, but it also has several features which reduce accidental erasure of files, something which is easily done with MINIDOS-PLUSX. (Although, for the sophisticated user, these same features may occasionally get in the way because they won't allow him to do some unorthodox operations which the DOS considers harmful.)

#### MINIDISK+ Operation

MINIDISK+ assigns each disk a name. All files can be identified either by a drive number, or by a disk name. For instance, in MINIDOS-PLUSX file DATA on drive 2 would be called 2/DATA; in MINIDISK+ it is called :2 DATA. But if the name of the disk is ABCDEF, then you can also access the file as :ABCDEF DATA. MINIDISK+ will search all active drives until it finds disk ABCDEF, and then use that disk.

MINIDISK+ also uses the "wild card" character /. The / is like a joker in a card deck - it can replace any other character or characters. Thus / all by itself stands for any file name; TEST/ stands for any file name which starts with the letters TEST, such as TEST1 or TESTER.

Percom's MINIDOS and MINIDOS-PLUSX assign a file type to files, thus differentiating between text files, Basic files, and plain program files. But virtually no software checks the file type. Though this may give an occasional weird result, it never does any damage, and does in fact allow you to do some unusual things. MINIDISK+, on the other hand, assigns file types and checks them carefully.

MINIDISK+ can handle up to 45 files per disk, rather than the maximum of 31 which MINIDOS-PLUSX handles. Moreover, MINIDOS-PLUSX uses sectors 000 and 001 of each disk for the directory, leaves sectors 003 through 009 blank, and starts file storage at sector 010; MINIDISK+ uses 000-002 for the directory, and starts file storage at 003, giving 347 usable sectors on a disk instead of only 340 (or 397 instead of 390 on a 40-track drive.)

Since MINIDISK+ occupies 2K, compared with the 1K of MINIDOS-PLUSX, it contains more commands. For instance, functions to copy files or pack a disk are part of the MINIDISK+ in EPROM, whereas they are utilities which have to be loaded from disk when using MINIDOS-PLUSX. This does make operation a lot smoother.

In the following list of MINIDISK+ commands, the name of the closest MINIDOS-PLUSX command or utility is enclosed in parentheses after the MINIDISK+ command, although there are a lot of subtle differences between them. (All commands can be abbreviated to two letters.)

NEW (I) initializes a new disk, and gives it a disk name.

SAVE (S) saves memory contents to disk. If a file with the specified name is already on disk, MINIDISK+ checks whether you want to replace the old file. After the file is written, MINIDISK+ reads it back to check that it was written properly. Many users will find this read-after-write capability invaluable.

LOAD (L) loads files back into memory. A disk and/or file name can be specified, or the DSSS format can be used, as in MINIDOS. In either case, an alternate target address can be specified, so that a file may be loaded into a different location from that it was saved from.

ADD (A) adds 10 sectors to the end of a file to allow for future expansion.

OPEN (CREATE disk utility) creates an empty file on the disk.

REMOVE (D) removes one or more files from the disk. As in MINIDOS-PLUSX, the file space from a

removed file gets tacked on to the previous file, but with a difference. The vacated space is listed in a disk directory printout as available, and subsequent SAVE commands will insert other files into this empty space if they fit, permitting this space to be reclaimed without repacking the disk. This feature provides many of the advantages of dynamic sector allocation as used in Flex and other disk operating systems, without some of its disadvantages.

CHANGE (R) renames a file. If the new name already exists on the disk, then an error message is printed. There is no provision for file protection.

ANALYZE (F) prints the disk directory. Empty holes left by previous REMOVES are identified, and a total of the available area at the end of the disk is also printed (unfortunately in hexadecimal; there obviously wasn't room for a decimal conversion routine in the EPROM.) The ANALYZE command can be followed by one or more file names (with or without the wild-card character /), in which case only the specified files will be printed.

RUN, followed by a file name, loads a program file and runs it.

GOTO (J) jumps to a program in memory.

EXIT (X) returns to the monitor.

SQUASH (PACK disk utility) packs a disk to eliminate holes. Its error-handling appears much better than that of Percom's PACK utility and is less likely to clobber a disk if something goes wrong. Each time a file is moved on the disk, it is read before and after the move to check for errors. If an error is detected, the file name being processed is printed, and the directory is rewritten to indicate the current status of the disk.

COPY (COPY disk utility) copies files from one disk to another. In some ways, this command is much more powerful than Percom's COPY utility. Files can be copied from one drive to another, or to the same drive; in the latter case, MINIDISK+ will ask you to SWAP disks, thus allowing copying disks on single-drive systems. A whole series of file names can be specified in the command, or the wild-card character / can be used, so that more than one file can be copied, or in fact, an entire disk can be copied. If a file name already exists on the new disk, MINIDISK+ asks whether the old file should be removed. COPY also checks a file after it is copied to reduce errors. (There is no BACKUP routine furnished, but Percom's BACKUP will copy MINIDISK+ files very nicely. Copying a disk is preferable to doing a BACKUP, though, since it will also repack a disk if it has vacant space.)

Although the COPY command could be used to copy a disk having a MINIDOS-PLUSX directory into a MINIDISK+ directory, this function is more easily handled by a utility called CONVRT.

SIZE prints out the amount of contiguous memory available on the system. This function is used by several other commands to determine available memory size for buffers, but can also be used as a quick memory test.

CK reads a disk file and checks it for errors. The file is read into the area normally used to hold the directory, and does not disturb other memory. Thus it can be used to test a file after writing, so that it can be re-written if an error exists.

#### MINIDISK+ Supporting Software

A DOS is not much good without software to go with it. What is available?

First of all, all Percom software which uses the DSSS format will still run. This includes the Touchup Editor, Text Processor, and Percom Assembler.

Cer-Comp also provides several disks of programs or patches to other programs to adapt them



to MINIDISK+.

Patch Disk No. 1, at \$19.95, is an almost essential investment. It patches a number of programs to run under MINIDISK+; the patched programs then allow using named files for data and programs. Some of these are very well done.

The patch to SWTP Cores Editor/Assembler is a tremendous improvement over the original, or even over Percom's patch. It allows both source as well as object programs to be saved on disk; it also provides a memory option, which allows the object program to be directly placed into memory for execution. This patch also fixes up some Cores problems, and makes it into a really first-class assembler. The only limitation is that Cores can only assemble a source program which fits into memory.

Many old-timers - Is there such a thing? - who have moved up to a TSC or Percom assembler tend to look down on Cores as old-fashioned or unreliable. It takes a while to realize that with Cer-Comp's patches, Cores is suddenly transformed into a completely new and very useful program.

Another patch is for SWTP 8K Basic versions 2.0 or 2.2 which allows Basic programs to be loaded and saved on disk. It will not, however, load Basic programs from either Percom Super Basic, or from earlier Percom patches to the same SWTP Basics.

The patch disk also has two patches to the cassette version of Microware's A/BASIC compiler. The first of these allows A/BASIC to compile a disk source file into a disk object file; this patch is roughly equivalent to the A/BASIC patch on one of Percom's user group disks. The second patch allows A/BASIC to operate with disk files, a very useful addition. Another patch (on Disk No. 2) allows A/BASIC to be coresident with an editor, so that programs can be edited, compiled, and either stored on disk, or immediately placed into memory for execution.

The same disk also has a patch for Percom Super Basic to allow it to work with MINIDISK+. Though it makes some minor changes to commands, Basic operation appears unchanged. However, where the original version of Super Basic ignores file types, Cer-Comp's patched version does not. Hence Super Basic can no longer read text or program files as data and operate on them (unless patched). This disk also includes a conversion utility to convert disks with Percom MINIDOS-PLUSX directories into the MINIDISK+ format.

Cer-Comp's Disk No. 2, at \$39.95, contains more useful software. It includes a patch to the SSB Source Generator to allow disassembly and storage of text on disk, as well as programs to list disk files, and relocate program files when loading them. The latter adjusts extended addresses within the program to allow machine language programs to run after they are relocated. Also included is a disk-to-disk assembler which will handle programs too long for Cores, as well as the patch to permit coresident operation of ABASIC with Cer-Comp's Editor. Another utility permits reconstruction of a disk directory in case of an accidental erasure.

The Editor itself is also on this disk. Although it takes a bit of getting used to, it is a fascinating program to watch at work. In addition to the more common editor functions, such as string search and replacement, line replacement, and so on, Cer-Comp's editor has a line editing function which is sorely missing from TSC's editor, used in other 6800 disk systems. When characters are added or removed inside a line, the editor quickly re-writes the rest of the line and then backspaces the cursor to the character being worked on. In this way it always shows the line as it currently exists, and opens up or closes spaces within it. This function obviously requires a CRT terminal, and uses control characters for a CT-64 terminal; but the manual shows where the various control characters are

located so they can be changed for other terminals (although Hazeltine or other terminals which require a combination of two control characters for each function might require extensive patching.)

The editor also has a renumber facility for Basic programs, and does file conversion between Cores or SWTP Basic program format, and Percom Super Basic or Percom Assembler format.

#### What it Costs

MINIDISK+, including manual and two 2708 EPROMs costs \$69.

Patch Disk No. 1, at \$19.95, is an almost indispensable addition, since many of the features of MINIDISK+ are not utilized with programs which only use the DSSS format commonly used by Percom software.

Disk No. 2, at \$39.95, contains the Cer-Comp Editor, and is also a very important addition.

Thus the minimum system price for a new user would be about \$89 for MINIDISK and Patch Disk No. 1. The complete package of all software, attractive for the user who is switching from MINIDOS-PLUSX and has some Percom software as well, would be about \$129.

#### Conclusions

On the whole, MINIDISK+ and its supporting software works and works well. For a new owner of a Percom LFD-400 disk system, it provides an attractive alternative to Percom's MINIDOS-PLUSX.

Although MINIDOS-PLUSX provides for named files, most Percom software does not support that ability and still uses the DSSS address format for specifying file location on a disk. The disadvantage is that a slip of the finger can put a file in the wrong place on a disk and erase something.

MINIDISK+ and its software supports named files, so there is much less chance for error. Add to this its error checking, and you get a system which is much less likely to clobber a disk or erase valuable data. The result is an idiot-proof system more like the 'big' DOS systems.

MINIDISK+ also provides a number of new, ROM-based commands which do things which require disk-based utilities elsewhere. This too makes operation easier and more convenient.

In a way, MINIDISK+ provides some of the niceties that are present in TSC's FLEX or in the SSB DOS, but without their slow speed and without requiring two drives or large amounts of memory or disk space to hold utilities.

On the other hand, a more experienced user, or one who has a heavy investment in Percom software, may feel differently about it. He may not need all the features of MINIDISK+, or may find that it sometimes prevents doing some intentionally 'wrong' operation. Some may also be bothered by the need to convert existing disks, or to relearn a new system.

In any case, deciding on a DOS is a very important decision which affects every other use of the system for a long time to come. Cer-Comp's MINIDISK+ is an interesting alternative to consider.

Additional information may be obtained from:

CER-COMP Microcomputers  
5566 Ricochet Avenue  
Las Vegas, Nevada 89110  
(702) 452-0632

\* CORRECTION \*

The February '80 issue carried our 1979 Index. An error snuck in! Page 11, the 'Christmas Card Programs' was written by Chaplain (Maj) USA, Paul E. Phelps.

## BASIC UTILITY PACKAGE

Dale Puckett  
14753 Endsley  
Woodbridge, VA 22193

This month we review six utility programs designed to make the life of the BASIC programmer a little easier. They are offered by Star-Kits, P. O. Box 209, Mt. Kisco, N. Y. 10549. One of the programs is written in assembly language, the others are coded in BASIC.

The utilities are: BASEDIT, an editor designed mainly to renumber BASIC programs; PRETTY, a pretty-printer; VINDEX, a program which indexes variables; BACOMP, a utility which lists the differences between two BASIC programs; SHORTS, which shortens listings and speeds up execution of some programs; and BENTER, which automatically generates line numbers and puts a program on a disk. A bonus utility called FLOGEN also comes with the package. Given enough time it will print a flowchart of a BASIC program, pointing out all FOR-NEXT loops and transfers of control.

The packages are available in three versions: MF runs on the Mini-Flex DOS and SWTPC Disk Basic Version 3.0; F2 runs with Flex 2.0 and TSC Disk Basic; and PD runs with a Percom LFD-400 disk, Minidos-PLUSX DOS and Percom Super Basic. Some of the programs require 32K of memory although they can be modified to use less memory.

BASEDIT, the assembly language program is supplied in both source and object form. It was designed specifically to renumber BASIC programs and changes all GOTO's, GOSUB's, etc., within the program while it is changing the line numbers.

BASEDIT will prompt you for a starting line number. If you do not give one, it will start numbering lines at 1000. Unfortunately however, it will only increment line numbers by 10. This may seem like a shortcoming to the programmer who loves to remove all the bugs and then renumber his program with a line increment of one in order to make it hard to change.

BASEDIT is menu driven and also allows you to do minor editing to the BASIC file. It includes function to both (F)ind and (R)eplace strings. They both seem to work although editing with a full size editor is obviously more efficient. Both work on every occurrence of the string in the program and the user should proceed cautiously. Stark says in his well written documentation that BASEDIT might take several minutes to renumber a long program. Yet, it seems to be fairly fast and I timed it at 14 seconds on a program 32 sectors long.

Pretty, the pretty-printing program does several things. It separates the listing into pages, providing a page heading complete with date and page number on each sheet. It double-spaces before and after all REMs, breaking the listing into easily readable blocks. If a REM is encountered within a line it is automatically placed at column 50.

Pretty indents each statement in a FOR-NEXT loop thereby illustrating the range of the loop. Nested loops are indented further and a very readable program results.

The program prompts you for the Port number, the number of lines on a page, the program name, and the date. It seems a shame that an option to control the width of the listing was not provided.

There was no problem listing a program on my IBM typewriter, but when I tried one on my Model 15 teletype I occasionally ran out of space. I do not like to make routine listings on my IBM because it costs \$30 just to get an estimate for repair.

When using the mini-FLEX version you must change line 180 so it will know which file to open for read. If you do not make this change before running you will get an error message which points to the line, but, it seems it would be much nicer aesthetically if the author had prompted you to make the change and then type "CONT."

VINDEX prints a list which gives every line number where a variable is used. It is one of the most useful programs in the package. Its only fault appears to be its speed. I timed it at 11 minutes and 47 seconds before it started printing on the same 32 sector program mentioned earlier. It then took another four minutes to print the results on a CRT running at 60 characters per second. Most of this problem is caused by the fact that it is running in the original SWTPC BASIC. Here's the good news. On the documentation a stamp noted that an experimental version of VINDEX.COM was enclosed on the disk. I'm pretty sure that VINDEX.COM was compiled on the ABASIC compiler. It is fast and indexed the same 32 sector program in a little over 35 seconds, start to finish.

BACOMP is a utility that will help you find the changes you made in later versions of your BASIC programs. It reads from two separate BASIC program files and prints only the differences. Every time it encounters a line that is different it prints the line from one file and then indents and prints the same line from the other file.

SHORTS is a program you can run on your BASIC masterpiece after you have removed all the bugs. It shortens the program by removing all remarks that are not on lines referenced elsewhere and by concatenating several short lines into one. It is also possible to have it print you a list of all program transfers sorted by destination because the program needs that information before it can remove any REMs.

SHORTS is also hampered by the lack of speed of SWTPC BASIC. It took it just over 13 minutes to run on PRETTY, a BASIC source file which is 28 sectors long. But, the new file was only 16 sectors long.

BENTER is a short program and seems to perform as expected. It faithfully generated automatic line numbers while I typed in a short BASIC test program. And, it allowed me to pick the starting line number and the line increment. The resulting disk file loaded into BASIC and ran perfectly.

FLOGEN, the bonus program is also interesting. It reads a BASIC program from a disk file and prints it in an abbreviated form with arrows and lines connecting segments of code which go together. It also connects each NEXT with the proper FOR. It does not illustrate transfers caused by a GOSUB however since they almost always involve long transfers from one end of a program to the other. FLOGEN also runs slowly and it takes nearly 30 minutes to print a chart of VINDEX.BAS. VINDEX is 37 sectors long.

### CONCLUSIONS

For the person who spends most of his time writing and debugging BASIC programs, this Basic Utility Package should be well worth the money.



For those still using SWTPC type basics BASEDIT should be a great help. PRETTY is a program every BASIC programmer should be required to use. What good is a program if you can't read it six months later? PRETTY will go a long way toward improving your readability problems. VINDEK will shorten a lot of headaches, especially for those programmers who write long BASIC programs with many variables. You probably couldn't find an easier way to keep track of them. SHORTS will help you out by improving the speed of execution. Just make sure you save a copy of the original program with all the REMs.

The numbering and formatting are almost automatic and the result is extremely pleasing to the eye. Readability is the answer to many programming problems and Stark is making it possible for you to let the computer do this housekeeping chore while you worry about the problem you're solving. Isn't that why we use computers?

#### INTERFACING THE HEATHKIT H14 PRINTER TO THE SWTP 6809 SYSTEM

MILLIAM A. CONER  
419 BLACKMAN ST.  
LAKE CHARLES, LA 70605

THE HEATHKIT H-14 PRINTER IS, IN MY OPINION, THE BEST PRINTER VALUE ON THE MARKET. THIS ARTICLE GIVES THE REQUIRED HARDWARE, HARDWARE MODIFICATIONS, AND SOFTWARE CHANGES NEEDED TO INTERFACE THE H14 TO THE NEW SWTP 6809 COMPUTER AND TSC 6809 DOS.

ONCE YOU HAVE YOUR MP-09 BOARD AND 6809 DOS, THE NEXT REQUIREMENT IS AN ACIA BOARD INSTALLED IN SLOT #7. I RECOMMEND THE SI-1 BOARD BY MIDWEST SCIENTIFIC, BECAUSE IT HAS THE RTS AND CTS CONTROL SIGNALS BUFFERED AND CONNECTED TO A 25-D PIN CONNECTOR AND CAN BE DELIVERED IN 2-3 WEEKS. THE REQUIRED JUMPERS FOR THE SI-1 ARE LISTED IN TABLE S-1.

THE NEXT STEP IS TO COMPLETE ASSEMBLY OF THE H14 WITH MODIFICATIONS TO OBTAIN THE COMPLEMENT OF THE BUSY SIGNAL ON CONNECTOR PIN #4. THE INSTRUCTIONS FOR THIS MODIFICATION ARE INCLUDED IN YOUR HEATH MANUAL. SINCE THE HEATH 25-D CABLE SUPPLIED WITH THE H14 DOES NOT MATCH THE SI-1 25-D CONNECTOR IT MUST BE REARRANGED ACCORDING TO INSTRUCTION 1H TABLE S-2.

THE TSC DOS ALLOWS CUSTOM PRINTER DRIVERS TO BE EASILY ADDED BY REPLACING THE PRINT.SYS FILE ON THE SYSTEM DISK. THIS IS ACCOMPLISHED AS FOLLOWS:

- (1) CREATE A FILE CALLED H14PRN.TIT AND ENTER THE FOLLOWING DRIVER ROUTINE: \*\*\*\*\*H14PRN\*\*\*\*\*
- (2) ASSEMBLE DRIVER ROUTINE: \*\*\*\*\*ASMBR-H14PRN\*\*\*\*\*
- (3) LOAD H-14 DRIVER INTO MEMORY: \*\*\*\*\*GET-H14PRN\*\*\*\*\*
- (4) REPLACE TSC DRIVER WITH H-14 DRIVER: \*\*\*\*\*SAVE-PRINT.SYS,CCCB,CEE4\*\*\*\*\*

- \* THE SI-1 ACIA INTERFACE BOARD IS AVAILABLE FOR \$50 KIT, #87 ASSEMBLED
- \* \$3.00 FOR SHIPPING FROM:

MIDWEST SCIENTIFIC INSTRUMENTS  
220 WEST CEDAR  
ALATHE, ILLINOIS 60621  
913-764-3273

THE DETAILED ASSEMBLY AND OPERATION MANUAL FOR THE H-14 CAN BE OBTAINED FOR \$3.50 (REFUND IF H-14 IS PURCHASED) FROM: HEATH COMPANY, BENTON HARBOR, MI 49022

\* H-14 PRINTER DRIVER FOR PRINT.SYS OVERLAY FILE  
\* 6809 CODE  
\* PORT EQU \$E01C SLOT #7 PORT ADDRESS

```
*
ORG $C000
* INITIALIZE ACIA SBR
PIHIT LDX #PORT GET PORT ADDRESS
LDA #03 MASTER RESET ACIA
STA 0,X
LDA #11 CONFIGURE ACIA
STA 0,X
RTS
```

```
* CHECK FOR PRINTER BUFFER FULL (BUSY)
* H14 RTS WILL GO "HIGH" WHEN BUFFER CAN ACCEPT MORE
* H14 RTS -CONNECTED TO - SI-1 CTS
* CAPABLE OF UP TO 4800 BAUD RATE WITH "HANDSHAKING"
```

```
ORG $C000
POK: STX #0 SAVE X-REG
LDX #PORT PORT ADDR OF PRINTER
PCL LDB 0,X GET STATUS REG
ASRB SR BIT 0 TO C-FLAG
ASRB
ASRB
ASRB SR BIT 3 TO C-FLAG
BCS PCL WAIT FOR PRINTER BUFFER EMPTY
LDB #11 SET UP CALL TO 'POUT'
RTS
```

```
* OUTPUT CHAR TO H-14 BUFFER
```

```
ORG $C0E4
POUT STB #A2 SAVE B-REG
BSR POK: CHECK FOR BUSY
STB 0,X #11 TO CONTROL REG
PCL LDB 0,X GET STATUS REG
ASRB SR BIT 0 TO C-FLAG
ASRB SR BIT 1 TO C-FLAG
BCC PCL WAIT FOR TDR EMPTY
```

STA 1,X OUTPUT CHAR  
LDX #0 RECALL X-REG VALUE  
LDB #A2 RECALL B-REG VALUE  
RTS

\* H-14 PRINTER DRIVER FOR PRINT.SYS OVERLAY FILE  
\* 6809 CODE  
\* PORT EQU \$E01C SLOT #7 PORT ADDRESS

```
*
ORG $C000
* INITIALIZE ACIA SBR
PIHIT LDX #PORT GET PORT ADDRESS
LDA #03 MASTER RESET ACIA
STA 0,X
LDA #11 CONFIGURE ACIA
STA 0,X
RTS
```

```
* CHECK FOR PRINTER BUFFER FULL (BUSY)
* H14 RTS WILL GO "HIGH" WHEN BUFFER CAN ACCEPT MORE
* H14 RTS -CONNECTED TO - SI-1 CTS
* CAPABLE OF UP TO 4800 BAUD RATE WITH "HANDSHAKING"
```

```
ORG $C000
POK: STX #0 SAVE X-REG
LDX #PORT PORT ADDR OF PRINTER
PCL LDB 0,X GET STATUS REG
ASRB SR BIT 0 TO C-FLAG
ASRB
ASRB
ASRB SR BIT 3 TO C-FLAG
BCS PCL WAIT FOR PRINTER BUFFER EMPTY
LDB #11 SET UP CALL TO 'POUT'
RTS
```

```
* OUTPUT CHAR TO H-14 BUFFER
```

```
ORG $C0E4
POUT STB #A2 SAVE B-REG
BSR POK: CHECK FOR BUSY
STB 0,X #11 TO CONTROL REG
LDB 0,X GET STATUS REG
ASRB SR BIT 0 TO C-FLAG
ASRB SR BIT 1 TO C-FLAG
BCC PCL WAIT FOR TDR EMPTY
STA 1,X OUTPUT CHAR
LDA #A2 RECALL X-REG VALUE
LDB #A2 RECALL B-REG VALUE
RTS
```

#### 0 ERROR(S) DETECTED

#### SYMBOL TABLE:

POK	CCB	PCL	CCD5	PIHIT	CCCB	PCL	CCB	POUT	CEE4
PORT	E01C								

THE H-14 25-D CONNECTOR MUST BE MODIFIED FOR USE WITH THE SI-1 25-D CONNECTOR. TABLE S-2 LIST THE REQUIRED MODIFICATIONS.

\*\*\*\*\* TABLE S-2 \*\*\*\*\*

H-14 S-2 CONNECTOR	HEATH 25-D	(ORIGINAL) COLOR	(MODIFIED) COLOR
PIN #2 PS-25C OUT	PIN #2	RED	BLACK
PIN #4 UN-CONNECT	PIN #15	GREEN	BLACK-GREEN
PIN #5 PS-25C IN	PIN #3	ORANGE	RED
PIN #7 PTS	PIN #4	BLACK-WHITE	BLACK
PIN #9 CTS	PIN #5	BLACK-GREEN	BLACK
PIN #15 ALSO	PIN #8	WHITE-BLUE	BLACK

HEATH INSTRUCTIONS TO MODIFY H-14 TO OBTAIN COMPLEMENT OF BUSY SIGNAL ON CONNECTOR PIN #4 MUST BE DONE AT ASSEMBLY TIME. BLANK = WIRE IS PERMANENTLY REMOVED.

\* LOCATED ON H-14 MAIN CIRCUIT BOARD

\*\*\*\*\* TABLE S-1 \*\*\*\*\*

JUMPER PAIRS ON THE SI-1 ARE TO BE AS FOLLOWS:

JUMPER PAIRS 1 AND 2 - INSTALL A JUMPER  
JUMPER PAIRS 3 AND 4 - OPEN  
JUMPER PAIRS 5 AND 7 - INSTALL A JUMPER  
JUMPER PAIRS 6 AND 8 - OPEN  
JUMPER PAIRS 9 AND 10 - OPEN  
JUMPER PAIR 600 JUM - INSTALL A JUMPER (4800 BAUD)  
NOTE: SWTP 6809 BOARD MUST BE PROPERLY CONFIGURED FOR 600 BAUD JUMPER TO SELECT 4800 BAUD. SEE TABLE MP-09

\*\*\*\*\* TABLE MP-09 \*\*\*\*\*

THE FOLLOWING OPTIONS ARE REQUIRED ON SWTP MP-09 6809 BOARD:

SELECT THE LOW BRID(ON) OPTION FOR SWITCH S-1  
SELECT 4800 ON THE 4800-6000 2-POSITION PROGRAMMING STRIP

2021 North Mall  
North East PA 16420

Don Williams Sr.  
Publisher: 68 Micro Journal  
3018 Hamill Rd.  
Houston TX 77343

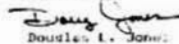
Don:

Thanks for your letter of May 30 describing 68 Micro Journal. It sounds like a good magazine one that I know I would enjoy. When funding allows, I perhaps will send in for a subscription.

Here is a small article that I hope you will be able to use.

STUFF describes a bunch of small utilities for the Altair 680 B. MITS supposedly sold over 2000 of these units. To date, I know of only 4 other people with them. Oh well! right right, would you.

I do hope you will send a copy of the magazine should you see the article.

Sincerely,  
  
 Douglas L. Jones

## STUFF FOR 680B

Douglas Jones  
 2271 North Mill  
 North East, PA 16428

STUFF is for the owner of a MITS Altair 680 B. (A proud but vanishing minority group.) STUFF bridges the gap of so many things missing for so many months. It has valuable utilities with the loader / dumper routines; it provides some of the MIKBUG utility routines; STUFF gives you a line-trace for CSAVE BASIC. Finally, for those distant relatives of the B-100 bus: (so near and yet so far away), it's something you can buy. Fill those empty ROM sockets! Gain valuable software and hardware experience!! Reminise about the good old days when this small wonder of the computer world sold for a mere \$292. in kit form!!

The hardware requirements are that you have the Altair 680 B: (either front panel or turn-key model); the 16 K memory expansion; and the KCACR card. STUFF will enhance most all of the software you are running on your system. When you put your STUFF ROM in the socket, new worlds will open up to you. BASIC and ASSEMBLER will be loaded with record breaking speed. You can amaze your friends and yourself with line tracing through BASIC.

The loader / dumper routine are why STUFF was written. The remaining routines were added in a frantic effort to fill the 1702 ROM with exactly 256 bytes of goodness no more, no less, or NOPs. The original paper-tape version of BASIC took some 45 minutes to load with a good 33 teletype! each minute one of those

until the final B9 signal of success. The KCACR cassette version of CSAVE BASIC (V 1.1 R 3) is quick to load, noiseless, and accurate! still it takes some 14 minutes to load. STUFF will cut this load time in half, plus give you an added bonus of jumping to an starting these CSAVE BASIC for you.

There will now be exactly 3 ROMs in the computer: the PROM MONITOR (address \$FF00); the KCACR MONITOR (\$FD00); and now STUFF (\$FE00). STUFF's loading address is \$FE00; while the dumper address is \$FE74.

The KCACR format, although at 300 baud, is still the Motorola tape format:

51BBAABDCC  
 59

51 Header record  
 00 The number of bytes  
 AAAA The address  
 DD The data  
 CC The checksum  
 5E End of file record.

The address, although representing only two bytes of data is dumped in (four) hexadecimal ASCII bytes: the same with the data and the checksum. By dumping and loading these characters in binary, there will be half the number of characters to dump and consequently half the load time is required.

The new binary format is simple:

M M B AA dddd... C  
 M M B JJ C  
  
 M Block mark  
 B Number of bytes  
 AA 16 bit address  
 dd Binary data  
 C Checksum  
 JJ 16 bit jump address

Each character is a full 8-bit binary character. After all you are loading with an 8-line! who cares whether you can print these characters or not?

The use of the loader / dumper is easy. Two examples will be given to clarify the dumper function.

### EXAMPLE 1

Let's assume you have a small assembly language program at \$2000 to \$2234 with a starting address of \$2123. You want to dump and quick load this program.

Load the assembly language program in your previous standard method. Start the dumper program and answer the following three prompts: starting address, ending address, jump address. For the above example it will look something like this:

.J FE74 72000 72234 72123

At the return of the second period the dump will have been made to cassette tape. Reloading and subsequently jumping to and starting this program is now simple. Just type: .J FE00 and the loader will do everything else.

### EXAMPLE 2

CSAVE BASIC lies in the area of \$0000 to \$1C1B, with a start address of \$0000. But there is a peculiarity in dumping it in that all of the monitors stack areas lie in the area of \$00F0 to \$00FF and cannot be dumped and successfully reloaded. This program (after normal loading but not initializing) must be dumped in two parts as follows:

.J FE74 00000 700E6 7FE00  
 .J FE74 70100 71C1B 70000

Notice how the first dump has a jump address folding back to the loader routine; the second jump is to the CSAVE BASIC start address. Loading your new tape back in is as above:

.J FE00

MEMORY SIZE 7

You see! It has loaded and started CSAVE BASIC.

Any number of program segments can be similarly dumped and loaded using the fold-back technique. When dumping, just point the jump address back to loader; the final jump is, of course, the start address.

The Motorola MIKBUG routines, as mentioned before, were added merely to fill space on the ROM. The routines include OUTEEE;



INCEE, and PDATA1. The subroutine call addresses certainly are not the same as the M1KBUD ROM, but the routines are available to you now. Please check the accompanying listing for the call addresses and the use of the routine. They are simple and almost self explanatory.

The CSAVE BASIC line trace routine does require a bit more explanation in its use. First, prepare a special copy of CSAVE BASIC: load in, but do not initialize the tape. Modify the POLCAT call to point to TRACE as follows:

```
.M 0627 FF FE
.M 0628 24 D0
```

And the banner: (this part is not necessary . . . Just nice)

```
.M 18C4 34 5B
.M 18C5 38 54
.M 18C6 30 57
.M 18C7 20 41
.M 18C8 42 53
.M 18C9 41 49
.M 18CA 51 42
.M 18CB 49 5B
.M 18CC 43 20
.M 1C16 C5 45
.M 1C17 00 87
```

Now create a tape using the double-dump method described above. The special tape now contains (TRADIC), the single new line-tracing version of CSAVE BASIC.

To turn the trace on: POKE 242,0

To turn the trace off: POKE 242,3

These moves can be done either in a command mode or done prospectively. (Please refer to the banner run). Note how a trace is provided at the beginning of each line and at the multiple statement indicators.

**SPECIAL NOTE**

A quantity of burned-in 1702 ROMs are available from the author at a postage paid cost of \$15. If you choose not to purchase at this time, you can still assemble STUFF to operate in your AM at address \$4300.

#### SAMPLE RUN OF BASIC LINE TRACE

```
LIST
5 INPUT "TRACE ON (Y/N) IF LEFT(Y+1) = 'Y' THEN POKE 242,0
10 FOR N=1 TO 10
20 PRINT "N = "N
30 IF N = 7 THEN GOSUB 100
40 NEXT N
50 POKE 242,3
60 END
100 X = N
110 Y = N: Z = N
120 RETURN
OK
```

```
RUN
TRACE ON ? NO
N = 1
N = 2
N = 3
N = 4
N = 5
N = 6
N = 7
N = 8
N = 9
N = 10
OK
```

```
RUN
TRACE ON ? YES
[5] [10] N = 1
```

```
[20] [30] [10] N = 2
[20] [30] [10] N = 3
[20] [30] [10] N = 4
[20] [30] [10] N = 5
[20] [30] [10] N = 6
[20] [30] [10] N = 7
[20] [30] [100] [110] [110] [30] [10] N = 8
[20] [30] [10] N = 9
[20] [30] [10] N = 10
[20] [30] [40]
OK
```

```
00001          NAM          STUFF
00002          **
00003          * BINARY LOADER - DUMPER
00004          * AND OTHER THINGS
00005          * VERSION 2.1
00006          * JUNE 1979 DLJ
00007          **
00008          00F4 LASADR EQU $F4 HIGH ADDRESS
00009          00F6 JMPADR EQU $F6 JUMP ADDRESS
00010          00FA XH1 EQU $FA STORAGE FOR X-REG
00011          00FB XLOW EQU $FB
00012          00FC BYTES EQU $FC BYTE COUNT
00013          00FD BEGADR EQU $FD LOW ADDRESS
00014          F010 STATUS EQU $F010 KCACR STATUS
00015          F011 DATA EQU $F011 KCACR DATA
00016          **
00017          * ROUTINES
00018          **
00019          * THE FOLLOWING ARE IN PROM MONITOR
00020          **
00021          FF81 OUTCH EQU $FF81 OUT CHARACTER
00022          FF00 INCH EQU $FF00 IN CHARACTER
00023          FF24 POLCAT EQU $FF24 POLE FOR CHAR INPUT
00024          FF82 OUTS EQU $FF82 OUT A SPACE
00025          FF6D OUT2H EQU $FF6D OUT 2 HEX CHARACTERS
00026          FFDB RESET EQU $FFDB
00027          **
00028          * THE FOLLOWING ARE IN KCACR MONITOR
00029          **
00030          F0DB BADDR EQU $F0DB BUILD ADDRESS
00031          F0F5 KOUTCH EQU $F0F5 OUT CHARACTER
00032          F0C5 LEADER EQU $F0C5 RUMP SOME LEADER
00033          **
00034          FE00          DRD          $FE00
00035          **
00036          * LOADER
00037          **
00038          FE00 CE FE09 B1 LDX $LDR SET JMP TO LOAD
00039          FE03 DF FA STX JMPADR TAKE JUMP
00040          **
00041          FE05 DE FA LD LDX JMPADR LOAD JUMP ADDRESS
00042          FE07 4E 00 JMP X TAKE JUMP
00043          **
00044          FE09 8D 34 L R BSR BYTDET GET FIRST BYTE
00045          FE0B C1 FF CMP R $FF RUBOUTS
00046          FE0D 26 FA BNE LDR PACK IT NOT
00047          FE0F 8B 2E BSR BYTDET GET NEXT BYTE
00048          FE11 C1 FF CMP R $FF BLOCKMARK
00049          FE13 26 F4 BNE LDR BACK IF NOT
00050          FE15 4F CLR A ZERO CHECKSUM
00051          FE16 8D 27 BSR BYTDET GET BYTECOUNT
00052          FE18 C0 02 SUB B $2 ADJUST
00053          FE1A D7 FC STA B $BYTES BYTECOUNT
00054          FE1C 8D 21 LD11 BSR BYTDET GET HIGH ADDRESS
00055          FE1E D7 FA STA B XH1 STORE 17
00056          FE20 8D 1D BSR BYTDET GET LOW ADDRESS
00057          FE22 D7 F8 STA B XLOW STORE IT
00058          FE24 DE FA LOX XH1 LOAD X WITH ADDRESS
00059          FE26 8D 17 BSR BYTDET GET DATA BYTE
00060          FE28 7A 00FC DEC BYTES DECREMENT BYTE COUNT
00061          FE2B 27 09 RED LDI3 DONE WITH THIS BLOCK
00062          FE2D E7 00 STA B X STORE DATA
00063          FE2F E1 00 CMP B X CHECK IT
00064          FE31 26 09 BNE BUN MEM NO GOOD
00065          FE33 0B INX BUN POINTER
00066          FE34 20 F0 BRA MORE GO BACK FOR MORE
00067          **
00068          FE36 4C LD15 INC A INC EMENT CHECKSUM
00069          FE37 27 CC LLD BEO LD ALL OK
00070          **
00071          FE39 7E F82C JMP $F82 CHECKSUM ERROR
00072          FE3C 7E FD2F JMP $FD2 MEMORY ERROR
00073          **
00074          FE3F F6 F010 BYTGET LPA B STATUS GET KCACR STATUS
00075          FE42 56 RDR B GET INTO CARRY BIT
00076          FE43 25 FA RCS $BYTGET BACK AGAIN IF BET
00077          FE45 F6 F011 LPA B DATA GET 8 BIT CHARACTER
00078          FE4B 1B ABA ADD TO CHECKSUM
00079          FE49 39 RTS
00080          **
00081          * DUMP ROUTINE
00082          **
00083          FE4A B1 F0C5 OUTRY JSR LEADER PUNCH LEAD
00084          FE4D C6 FF PUN0 LDA B $FF BLOCK MARK
00085          FE4F 8D 3E BSR PUNCH1 PUNCH TWICE
00086          FE51 8D 3C BSR PUNCH1
00087          FE53 96 F5 LDA A LABADR+1 SUBTRACT LOW ORDER
00088          FE55 90 FE SUB A BEGADR+1
00089          FE57 D4 F4 LDA B LABADR SUBTRACT HIGH ORDER
00090          FE59 D2 F0 SBC B BEGADR
00091          FE5B 26 04 BNE PUN2 NOT DO IT?
00092          FE5D 81 40 CMP A $40 LESS THAN 64?
00093          FE5F 25 02 BCS PUN3 BRANCH IF DONE
00094          FE61 86 3F PUN2 LDA A $3F NO, SO PUNCH 63
00095          FE63 97 FC PUN3 STA A BYTES
00096          FE65 8B 04 ADD A $4
00097          FE67 16 TAB $4 ADJUST 8 BYTES
00098          FE68 8D 25 BSR PUNCH1 MOVE TO B-REG
00099          FE6A CE 00FD BSR PUNCH1 PUNCH IT
00100          FE6D 8D 1C BSR PUNCH PUNCH ADDRESS
00101          FE6F 8D 1A BSR PUNCH
00102          FE71 20 03 BRA PUN3A
00103
```

```

00104      ** ENTRY LINE FOR PUNCHING DATA
00105      **
00106      FE73 01      NOP      LINE UP ENTRY LINES
00107      FE74 20 1F      BRA      GO TO PUNCH RUNTIME
00108      0
00109      FE76 DE FB      PUMJA LDX      BEADNR      GET POINTER
00110      FE78 BD 11      PUM4  BSR      PNCM
00111      FE7A 7A 00FC      DEC      BYTES
00112      FE7D 2A F8      BPL      PUM4
00113      FE7F 0F FB      STX      BEADNR
00114      FE81 43      COM A
00115      FE82 14      TAB
00116      FE83 8D 0A      BSR      PNC 1
00117      FE85 09      DEK
00118      FE86 9C F4      CPX      LASADR
00119      FE88 26 C3      BNE      PUM0
00120      FE8A 39      RTS
00121      0
00122      FE8D E4 D0      PNCM LDA B      X
00123      FE8D 1B      ABA
00124      FE8E 0E      INK
00125      FE8F 7E FDF5 PNCM1 JMP      KOUTCH
00126      0
00127      FE92 7E FDD8 DET JMP      RADDR
00128      0
00129      FE93 8D F8      GET3 BSR      GET LOW ADDRESS
00130      FE97 BF FD      STX      BEADNR
00131      FE99 8D F7      GET2 BSR      GET HIGH ADDRESS
00132      FE9B DF F4      STX      LASADR
00133      FE9D 8D F3      GET1 BSR      GET JUMP ADDRESS
00134      FE9F DF F6      STX      JMPADR
00135      0
00136      FEA1 8D A7      JSP      BSR      OUTAT      PUNCH IT
00137      0
00138      FEAC CE 00F6 JSP1 LOK      JMPADR      DET JUMP ADR ADR
00139      FEAD DF FD      STX      BE ADR
00140      FEAE 0B      INX      BUMP IT
00141      FEAF DF F4      STX      LASADR
00142      FEAB 8D A0      JSP2 BSR      PUM0      GO PUNCH IT
00143      FEAD 8D FDC5 JBR      LEADER      PUNCH FINISH LEADER
00144      FE90 7E FFD0 JMP      RESET      ALL DONE
00145      **
00146      0 MIKBUO CALL AT SEIDI
00147      0 OUTPUT ONE CHARACTER IN A-REG
00148      **
00149      FEB3 37      OUTEEE PSH B
00150      FEB4 16      JAB
00151      FEB5 8D FF81 JBR      OUTCH
00152      FEB8 33      OUTE1 PUL B
00153      FEB9 39      RTS
00154      **
00155      0 MIKBUO CALL AT SEIAC
00156      0 INPUT ONE CHARACTER IN A-REG
00157      **
00158      FEBB 37      INEEE PSH B
00159      FEBD 8D FF00 JBR      INCH
00160      FEBE 17      JBA
00161      FEBF 33      PUL B
00162      FEC0 39      RTS
00163      **
00164      0 MIKBUO CALL AT SE07E (PDATA3)
00165      0 PRINT DATA POINTED AT BY X-REG
00166      0 UNTIL EOT (80A) IS REACHED
00167      **
00168      FEC1 8D F0      PDATA2 BSR      OUTEEE
00169      FEC3 0B      INX
00170      FEC4 A6 00      PDATA1 LDA A      X
00171      FEC6 81 04      CMP A      84      EOT?
00172      FEC8 26 F7      BNE      PDATA2
00173      FEC9 3A      RTS
00174      **
00175      0 CARRIA E RETURN / LINE FEED
00176      0 AND NULLS
00177      **
00178      FECD 0B      CRLFNL FCR      80D,80A,800,800,804
00179      FECC 0A      FECD 00
00180      FECE 00
00181      FECF 04
00182      **
00183      0 CBABE BASIC LINE TRACE
00184      **
00185      FEED 36      TRACE PSH A      SAVE REGISTERS
00186      FEED 37      PSH B
00187      FEED 06 F2      LDA B      242      TEST IS TRACE ON?
00188      FEED 26 25      BNE      TRACE1      BRAN H IF NOT
00189      FEED 96 BA      LDA A      08A      LINE NUMBER HERE?
00190      FEED 2B 21      BMI      TRACE1      BRANCH IF NOT
00191      FEED 86 58      LDA A      8'C      PRINT FORM
00192      FEED 8D 5D      BSR      OUTEEE
00193      FEED 8D 5D      JBR      81732      GO PRINT LINE NUMBER
00194      FEED 8D 5D      LDA A      8'J      FINISH FORM
00195      FEED 8D 5D      BSR      OUTE
00196      FEED 96 0B      LDA A      11      BUMP POSITION COUNTER
00197      FEED 4C      INC A
00198      FEED 4C      INC A
00199      FEED 4C      INC A
00200      FEED 97 0B      STA A      11      SAVE IT
00201      FEED 97 0C      SUB A      12      END OF LINE?
00202      FEED 25 0B      BCS      TRACE1
00203      FEED 7F 000B CLR      11      CLEAR COUNT
00204      FEED 7F 000B LDX      BCRFLNL PDATA1      LOAD M80 POINTER
00205      FEED 8D 0C      BSR      PRINT IT
00206      FEED 33      TRACE1 PUL B      RESTORE REGISTERS
00207      FEED 32      PUL A
00208      FEED 7E FF24 JMP      POLCAT      GO POLE
00209      0
00210      END

```

TOTAL ERRORS 00000

Dan Johnson  
7655 S.W. Cedarcrest St.  
Portland, OR 97223

When developing programs with a floppy disk system you inevitably find after a time that you have several copies of your object program distributed over several disks. These are often slightly different versions of the same program and it isn't always easy to tell which version is where. To solve this problem I developed the following two commands for SSB DOS48 (developed on version 4.2). The first 'SETVER' is used to set a version message into an object file (one file you can use the SET or RUN as FINB commands on including a 's' command files). The second 'RNDV' is used to show the version message.

SETVER works as follows:  
SETVER<input file-spec><output file-spec>  
The input file will be copied to the output file. Before to copy is initiated the prompt 'VERSION HEADER' appears on the console...enter a line

(i.e. VER 1.0 16-AUG-79) (use a version number and the date but now message can be used. This message is copied to the output file and then the input file is copied to the output file. If the header message contained any upper case 'B's' or 'D's' these will be converted to lower case to prevent them from being recognized as object code record marks. (Object files in SSB DOS are stored in the MINIBUS II binary record format which uses a 'B' or 'D' to indicate a start of record similar to how 'S1' or 'S0' are used in the MINIBUS hex format. The SETVER command also does one other thing as it copies the object file it eliminates any null characters between records. These null characters can be present when the input file was created by APPENDIX several smaller object files together. Thus under these circumstances it is possible for the output file to be shorter than the input file. Since the input and output files cannot have the same file-name you will probably want to use the @NAME command to end up with the proper output file-name.

RNDV is used to show the version message installed by the SETVER command. Syntax: RNDV<file-spec>

RNDV simply displays on the console as ASCII all of the specified file until a 'B' or 'D' is encountered in the file which indicates the start of the object code.

Both of these commands could be easily altered for use with FLEX if you keep in mind that FLEX uses a different object record format than SSB and adapt accordingly.

```

MAY/4800 1.21 0000  SETVER      SET VERSION
29-SEP-79 10:46:14 Page 11 Fore 1

641
642      WITH      SETVER
643      WITH      W1-B0
644
645 *****
646
647
648
649
650 *****
651
652 *****
653
654 *****
655
656 *****
657
658 *****
659
660 *****
661
662 *****
663
664 *****
665
666 *****
667
668 *****
669
670 *****
671
672 *****
673
674 *****
675
676 *****
677
678 *****
679
680 *****
681
682 *****
683
684 *****
685
686 *****
687
688 *****
689
690 *****
691
692 *****
693
694 *****
695
696 *****
697
698 *****
699
700 *****
701
702 *****
703
704 *****
705
706 *****
707
708 *****
709
710 *****
711
712 *****
713
714 *****
715
716 *****
717
718 *****
719
720 *****
721
722 *****
723
724 *****
725
726 *****
727
728 *****
729
730 *****
731
732 *****
733
734 *****
735
736 *****
737
738 *****
739
740 *****
741
742 *****
743
744 *****
745
746 *****
747
748 *****
749
750 *****
751
752 *****
753
754 *****
755
756 *****
757
758 *****
759
760 *****
761
762 *****
763
764 *****
765
766 *****
767
768 *****
769
770 *****
771
772 *****
773
774 *****
775
776 *****
777
778 *****
779
780 *****
781
782 *****
783
784 *****
785
786 *****
787
788 *****
789
790 *****
791
792 *****
793
794 *****
795
796 *****
797
798 *****
799
800 *****
801
802 *****
803
804 *****
805
806 *****
807
808 *****
809
810 *****
811
812 *****
813
814 *****
815
816 *****
817
818 *****
819
820 *****
821
822 *****
823
824 *****
825
826 *****
827
828 *****
829
830 *****
831
832 *****
833
834 *****
835
836 *****
837
838 *****
839
840 *****
841
842 *****
843
844 *****
845
846 *****
847
848 *****
849
850 *****
851
852 *****
853
854 *****
855
856 *****
857
858 *****
859
860 *****
861
862 *****
863
864 *****
865
866 *****
867
868 *****
869
870 *****
871
872 *****
873
874 *****
875
876 *****
877
878 *****
879
880 *****
881
882 *****
883
884 *****
885
886 *****
887
888 *****
889
890 *****
891
892 *****
893
894 *****
895
896 *****
897
898 *****
899
900 *****
901
902 *****
903
904 *****
905
906 *****
907
908 *****
909
910 *****
911
912 *****
913
914 *****
915
916 *****
917
918 *****
919
920 *****
921
922 *****
923
924 *****
925
926 *****
927
928 *****
929
930 *****
931
932 *****
933
934 *****
935
936 *****
937
938 *****
939
940 *****
941
942 *****
943
944 *****
945
946 *****
947
948 *****
949
950 *****
951
952 *****
953
954 *****
955
956 *****
957
958 *****
959
960 *****
961
962 *****
963
964 *****
965
966 *****
967
968 *****
969
970 *****
971
972 *****
973
974 *****
975
976 *****
977
978 *****
979
980 *****
981
982 *****
983
984 *****
985
986 *****
987
988 *****
989
990 *****
991
992 *****
993
994 *****
995
996 *****
997
998 *****
999
1000 *****

```

```

6104 F6A17E 1461 CHNBY1 LOK COUNT CHECK TRANSFER UNIT
6109 270A 1471 BEU CHK2 IF ZERO GO BCE WHAT THIS BY
610B 09 1481 DFX
610C FFA17E 1491 STX COUNT ELSE DEX COUNT
610F 2029 1501 BNR PUTBY1 AND WRITE BYTE TO OUTPUT F1
1511
6111 7D617D 1521 CHK2 TST FLAG START OF RECORD FLAG
6114 2711 1531 BEZ NOTCNT THIS IS NOT THE DATA BYTE C
6116 36 1541 PRN A
6117 8B04 1551 ADDA $# ADJUST CNT
6119 87617F 1561 STA A COUNT+1 SAVE ADJUSTED COUNT
611C 2403 1571 BEC
611E 7C617E 1581 INC COUNT
6121 32 1591 NOCV PUL A
61 7F617D 1601 CLR FLAG CLEAR THE FLAG
6125 2017 1611 BNR PUTBY1 WRITE DATA BYTE CNF
1621
6127 8147 1631 NOTCNT CMP A B'D TRAN FER ADDR RECORD?
6129 2607 1641 BNE NOO
612B C602 1651 LDA B SET UP COUNT
612D 7F617F 1661 STA B
6130 200C 1671 BNR PUTBY1 WRITE IT
6132 8142 1681 NOP START OF RECORD?
6134 2605 1691 BNE NOB
6136 7C617D 1701 INC FLAG SET FLAG
6139 2003 1711 BNR PUTBY1 WRITE IT

```

HAL/6800 1.21 613B SHOWV SHOW VERSION MESSAGE  
29-SEP-79 101441251 Page 31 Fore 1

Symbols Sorted by NAME:

#ABORT/6094	#CDFH/7783	DFH/7786	DFHORG/7780	DONE/608B
EXIT/60CA	FCB1/60E5	FIN/618B	GETVER/609E	ILLFN/60CD
#TIME/7289	#KEYCHK/6009	#LENGTH/010B	#HSS/60D3	#HORG/7280
#ODPH/7780	OK/609A	#OUTEE/7284	#GAPP/0009	#ODEL/0007
#DDIRT/000A	#DDIRT/000B	#GFREE/0000	#GREN/000B	GGG4R/0004
#GSDM/0001	GSRC/0006	GSREAD/0005	#GSRN/0012	#GSRM/0013
#GSMC/0003	#GSMRIT/0002	READCK/60AE	#SHOW/6080	START/6080
#XCS/001F	#XCT/001E	XBR/0026	XES/0001	KTC/0000
#XFN/0003	#XFS/0000	#XFSU/000E	#XFT/000C	#XLSU/0010
#XNS/0027	#XNT/0026	#XPS/0029	#XPT/002B	#XSD/002A
#XUN/0002	#ZADDX/72A3	#ZANCHK/729A	#ZCOLDS/7280	ZDIE/729D
#ZDONE/7283	ZFLSPC/7291	#ZDCHAR/7294	#ZDETHN/72A0	#ZGMCHR/7297
#ELTIME/7285	ZHDM/729C	#ZOUTH/72AF	#ZOUTHK/72AC	#ZOUTST/72A6
ZCTPSE/728F	ZWARNB/7283			

## BCD MULTIPLY

Eben R. S. Visser  
Software Eng, Metalogics, Inc.  
1156 S State 106  
Orem, UT 84057

Some of us lie awake nights trying to figure out how to do packed BCD multiplies. Others of us, more reasonable perhaps, lie awake wondering what strange variety of people lie awake working on programs. If you are in the former camp, an answer is at hand. If in the latter, you may never discover your answer. Maybe we're born that way.

We have been using a calculator chip from National Semiconductor to do arithmetic for us in a product we manufacture based on the Motorola 6800 chip. When we switched to a 6809, we thought that perhaps we could write a BCD multiply routine to beat the chip.

The result was the following program. The chip can do a multiply in around 37 msec. Most of our numbers needing multiplication are 4 digits. This program can multiply two 4-digit numbers in 13.8 msec. It can just as easily multiply two 50-digit numbers (though not as quickly!).

The subroutine is position-independent, reentrant, structured, modular, and may be interrupted at will. It requires no storage and leaves all registers unchanged, except CC. The

call to the subroutine is entirely transparent to the calling program except for the fact that the arguments are replaced by their product on the stack.

In the method of using the hardware stack, of saving the machine state, of creating a global area, and of sending arguments on the stack we did our best to implement Terry Ritter's standards for good software on the 6809. You who have read much 6809 literature will recognize Terry as the genius behind most of it. He is the author of several articles in "68 MICRO JOURNAL", as well as the author of the large white "MC6809 Preliminary Programming Manual". We wholeheartedly endorse his recommendations for ending up with good structured, modular, reentrant, recursive, etc. code.

This subroutine is structured. The flow of logic is in a straight line from the top to the bottom with an occasional loop or split (if-then construct). If this approach to programming yields code which is perhaps a byte or two longer than it might have been, one is nonetheless rewarded by code which is easier to understand, debug, document, modify, explain, and maintain.

You should find the subroutine easy to use. You simply push two packed BCD arguments of equal length, and put that length in the (A) register. You then do a call to the subroutine (LBSR BCDMPY for position independence). The product is right on the stack where the arguments were. All else is unchanged, as stated earlier.

The listing below contains both the subroutine AND a sample calling program to show you how you might call it. The sample calling program is not a part of the subroutine, and should be deleted when you use this code. We hope that you find it useful, and we would be glad to hear from you on your applications, uses, problems, or whatever.





```

1001 E6 02 MFLP LDB -X GET RIGHT HALF OF BYTE
1002 A6 02 GET LEFT HALF
1003 48 SHIFR 21 LEFT
1004 48 LSLA
1005 48 LSLA
1006 48 LSLA
1007 48 LSLA
1008 48 LSLA
1009 48 LSLA
100A AA E4 ORA -5 PUT (A) AND (B) SIDE BY SIDE IN (A)
100B 35 04 PULS 6
100C A7 A2 STA -1 WRITE PULSED BYTE
100D 6A 0C DEC MSTR52-10,5 CTRMSTR-1
100E 6D 0C TST MSTR52-10,5 CTRMSTR-1
100F 26 5A LMC MFLP GO BACK IF NOT
1010 35 7E RALS A B, D, E, F, V RESTORE REGS
1011 39 RTS END

```

0 ERROR(S) DETECTED

SIGNAL TABLE

ADDRESS	1002	1003	1004	1005	1006	1007	1008	1009	100A	100B	100C	100D	100E	100F	1010
ADDRESS	1002	1003	1004	1005	1006	1007	1008	1009	100A	100B	100C	100D	100E	100F	1010
PC/HLF	1002	1003	1004	1005	1006	1007	1008	1009	100A	100B	100C	100D	100E	100F	1010

## HELP

Help,

Would some kind person please assist me with information or experience in accomplishing the following:

1. Preventing multiple character displays when some CT-64 keys are struck.
  2. Correctly configuring CT-64 for MINI-FLEX.
  3. Hook up of Okidata 110 Bi-Directional printer.
- My system is SWTPC 20K, Qual MF68, MPLA.

Sincere Thanks,  
M.B. Ritchie  
2806 NW 54th Ave  
Gainesville, FL 32601

## CLASSIFIED ADVERTISING

Only one set Hemanways LINK68, RA6800ML, XREF68, STRUBAL+, EDIT68 used manuals for SWTPC 6800 on 8" diskette for sale \$150.00 (new \$450.00). Manfred Peschke 603-889-2196 (eves)

Percom disk controller card, software, blank disks. \$300 value only \$200. Steve Carter, 227 Railroad Ave., Rifle, CO 81650

Teletype Model 43 \$925, CT-64 \$250. Call Lew 1-615-3442914.

## Applevalley Day School, Inc.

Offering Our Own Business Software  
In SWTPC Disk Ver. 3.0

## MEK-D2 to S50 BUS

Paul E. Phelps  
111 Division St., 19  
King City, CA 93930

### IN THE BEGINNING

While stationed in Germany a few years ago, I got the yen to start getting involved in the new and exciting world of home econ computers. After considerable study and without a whole lot of certainty about what I was getting into, I ordered Motorola's MB-DC Evaluation Kit for what I assumed would be a simple hands-on introduction to the field.

It went together without as much as a whisper of difficulty and worked like a charm...except that those hex digits didn't mean much to my wife or my friends and, while I thought I was having fun, my wife wasn't so sure about it. If I was going to go much further with this not exactly cheap hobby, that had to change. Around my home, an unhappy wife usually meant great difficulties in spending money on an electronic project. So I began to think about upgrading to something more "workable."

I already owned a video terminal unit I had been using for some time on my amateur radio station and it happened to be ASCII as well. Now, if only one computer would speak to the other! What follows is a much abbreviated (I edited out the frustrated wails and tried to edit out some of the confusion that was involved along the way) attempt at a floor plan for converting the Motorola MEK-D2 Kit to work with SS 50 buss compatible boards. I might add at the beginning, that, as I am writing this on said machine, in a text editor and processor bought not too long ago, since you are reading this, I was successful.

In the first place, as silly as it may sound, build a mother board of some sort (or buy one already set up) and don't mess around with anything else. If you have already started using ribbon wire or something at a between the boards buss, STOP! I won't bore you with the detailed expenses of data buss or address buss buffers shot or broken wires, but a mother board is a must.

I ended up with a sheet of plastic about 14" by 14" drilled with rows of 50 holes each. I etched the SUP700 edge connectors to the board (on top - it was just thin enough) and used bare tinned corner wire beneath to create the buss. Almost radically lots of minor problems and irritations disappeared. I know I didn't invent it, but I felt as if I had created it from dust at the tenth wonder of the world - THE MOTHER-BOARD. Believe me, it was worth it.

### RAMELINGS

Cut a notch in one end of the plastic and mount an edge connector for the MD-02 board there. Wire the CPU board and the buss itself using insulated wire, as the lines cross over one another at this point. (It would have been too much to hope for that SUP700 and Motorola would use the same address buss and data buss layouts.)

### REMEMBRANCES

Wire the edge connector between the mother board and the MEK-D2, following SUP700's (or whatever board you are using) memory card pin locations and address/data buss from the Motorola schematic. Except for a couple changes, this is direct but can be confusing. Take your time!

The exceptions: Take the valid memory address (VMA) from pin 89 of U-89 (BT56), which goes to the data selector (U 81) (an MC741533) and wire it to the MD-100 address pin on the motherboard (pin 911 on the motherboard).[]

You will also need to make a change in the clock arrangement for the off-board memory systems. SUP700 and the SS 50 buss folk use a clock pulse that is inverted from the clock pulse provided by the MEK-D2 edge connector. I ran mine through an extra gate in an 8136 I was using for another purpose in the wire-wrap portion of the CPU board, then simply hooked it to a spare pin on the CPU edge connector. There are several unused pins available. This is the "Phase 2" or 82 clock on the Motorola schematic. But, since Motorola used non-inverting buffers in the kit, and SUP700 and the SS 50 folk use inverting buffers, you will need to invert this clock pulse. (It's pin 14 or 15 of U83 or 11 is also available from pin 83 of U815 - the clock itself?)[]

### THE MAIN BOARD

There are several changes that have to be made to the CPU board itself to prepare it for another monitor ROM and various other situations you will encounter. I will assume that you are going to use PERCOM's LDF-400 disk drive and will eventually want 32K of ram on line. I can't speak for the other disk drives, so research it if

- 1- IC destinations and pin numbers of IC's are Motorola numbers from the schematic that comes with the kit. You will also find that Motorola's Applications Note AN-721 will be extremely helpful in figuring out what you will want to do.
- 2- If you also buy PERCOM's Superbasic, an excellent BASIC in my opinion, you will need to set the end of memory limit manually prior to running BASIC. (On versions prior to 2.0.) Otherwise it cannot find the actual end of memory due to the non-inverting buffers. They explain in their manual.

you think you want another brand.

The problem centers around the fact that, probably for simplification purposes, Motorola decoded all ram locations above 3FFF as on-board locations only, using a 7438 (U87) to turn the data buss buffers on and off accordingly. However, the PERCOM LDF-400 monitor system resides at 0800h and needs scratch pad space from 8000 to about 7FFFh. If one wants to have 32K ram, that runs to 7FFFh, which is also above the 3FFFh cut off. To do this you will also give up the Motorola EPROM locations from 6000h up (or the sockets will have to have their addresses changed - they are hard wired at 6000h plus at present.)

I removed various lines from the 7438 (U87) and tied them high using, in the final analyses, only DBE (Data Bus Enable), R/W (Read/Write), and I/O (In/Out) to tri-state the data buss transceivers to tri-state. Additionally I decoded address lines A7, A8, and A9 with a 74135 (U3) and fed its output to the 7438 to provide a signal to turn the data buss off when the CPU was addressing the stack locations (8000 to 7FFFh). This also clears up the addressing problem with Mini-Dos Plus X, which is at 0800h, by placing that location also "off board" as far as the CPU is concerned through its data buss buffers.[]

At this point memory locations 8000h through 7FFFh, and 8000h through 7FFFh are seen as "off-board" by the CPU and, hopefully, all is well.

This latter was the most difficult for me, as the MEK-D2, to be used with the LDF-400 disk unit, must see the memory as above. (Especially since I want to be able to run at least 32K RAM.)

### SOME OTHER CONSIDERATIONS

I had already changed the clock speed of my MEK-D2 kit to 1 MHz prior to purchasing the LDF-400 disk drive, but if you haven't done so, you will have to. Order a 1 MHz clock module from Motorola.[] Install it in the clock module socket of the MEK-D2 kit. Cut the trace from the clock socket (pin 824, U813) that leads to the baud rate generator (MC 14080, U 817). I etched the old clock on the back of the new clock, after installing the new one in the socket. Wire it for 1500C and ground and take the 20C pin of the old clock (pin 824) and connect it to the U817 side of the trace you just cut. Also, pins 828 and 822 should be wired to 1500C so they will be held "high" for the old clock to

3- These can be picked up from pins 4,6, and 10 of U2.  
 4- Since the original of this article was written, I have added one of GIMM's 16K ROM boards and now have all ram except 8000-9FFF and E000-FFFF decoded as off board. In the final analysis, this was a lot simpler, if more expensive. U7 now only decodes "on-board" the I/O and the SUTBUG monitor ROM.  
 5- Order XLN 107295 K1122R 1.8 MC MC68018 unit from Motorola INC., Communications Systems Div., Consumer Products, 2535 Edsington St., Franklin Park, Ill. 60131.

work properly. That way you will have the proper pulse rate for the I/O devices (the ACIA is without having to do the more complicated planning of rates based on INUC and buy the new divider chips. You already have the old clock, I67

Protest: the I/O clock is fine and the processor is now running at 1MHz. It's faster (though not a great deal) and, more importantly, it is switched with the LFD-400 disk drive unit, which requires a 1MHz CPU speed.

Of course the 6818 IC at 8000h on the main CPU board takes care of SUTBUG's needs for scratch-pad ram. But if you buy the LFD-400 with Minidisk, the only way to do it is such more useful (that way) you will need a few bytes of ram from 8000h to 80FFh for Minidisk's scratch-pad area.

That I did was long and, perhaps, crazy, but worked. I had five extra 6818's from the MEK-D2 kit's original component and a backup. I ended up decoding the ram area above 8077h as "off-board" (as discussed earlier). Then I used the data bus portion of the on-board ram sockets so that they would appear to be "off-board" to the processor. That is, I cut the data bus traces to chips U014, U018, U019 and rewired the traces to the off-board sides of the data bus transceivers (pins 83,6,10, & 13 of U04 & 3). I added one additional 6818, hard wiring it in with ribbon wire. (7)

It makes for a messy CPU board this way. But it does work and it used parts I already had available and knew were good. Shortly I hope to correct this situation with another memory board and put 4K RAM at the 8000h location. Then I can remove all the extra wiring on the CPU board (9).

I should also note that Motorola uses a simplified form of addressing for the I/O ports on the CPU board. That is to say, they do not necessarily decode each device fully, relying on their own considerations that there won't be other ports on that board. If you end up doing what I did, and need extra ports on the board, it is a very good idea to be certain each device is fully decoded.

By way of example, the ACIA which Motorola decodes at 8000h as the cassette interface port, which I moved to 8080h as port 01, has one chip select line (pin 61) permanently wired to 15VDC. I removed it (cut the trace) and through an inverter (a 7404 in this case) hooked it to the 10 pin so it would not respond to 3014h (port 05) where I have a hardware clock hooked to a PIA. No further troubles noted. However, the problem surfaced when the software begins to "talk" to the clock board, de-initialized the ACIA and my terminal lost contact with the computer. It was a long and difficult problem to find. Hopefully the problem will not exist for you.

- 6- As far as I know, Motorola does not offer a trade in on the clock and the new one cost me \$24.00 in 1978.
- 7- Not a terrible good idea, but my vineyard area was already full and there wasn't much other choice at the time. Besides, it works!
- 8- See foot note 4.

Originally I cut out and mounted the key pad and LED display board on the hinged front panel of my case, but never use it now. However, I do use the cassette interface from time to time, and so would not recommend doing away with the entire board.

The box is an 8.5" x 12.5" x 19.5" steel box with an aluminum front panel, which I added hinges to. Unfortunately, it was purchased in Germany and I cannot suggest any place to buy a case for the completed unit. My own experience suggests that it won't be easy, as the MEK-D2 boards are a bit out-sized compared to the SS-50 boards.

I also used D-25 pin connectors between the box and the LFD-400 drive and a Heathkit Music Board I also have. I used 6-pin Molex connectors for other than 10 wiring and added an extra 18VDC plug (wall type) to the back panel for plugging the printer and drive power supplies in.

The power supply is a 25 AMP. transformer from Sunny International 193 with an 120V for the LFD-400 on the same transformer. I have 7000PF capacitance on the 800V line but only about 100PF on the 12 VDC line. So far I have experienced no power problems and the system seems to hold up even under somewhat low voltage conditions from time to time.

My present system is a 48K byte RAM, SUTBUG, LFD-400 with MINIDISK XL, cassette interface, and a video terminal unit at 1200 baud on an RS 232 type I/O port (at Port 01), a Heathkit H14 printer on Port 06 (six was easier to decode with what was available to me at the time). I use the spare PIA at 0014h for JPC's clock board for real time applications. I added an extra ACIA (MC 6550) at Port 6 (3018h) for the printer's serial port requirements (10).

In retrospect, there were two or three major crises for me. The first was figuring out with an absolutely archaic (Anti-Deluvian even) scope that the clock pulse had to be inverted for the SS-50 bus to work with this kit. That was in Germany and luckily my ravines in both German and English went unnoticed by the psychiatrist who lived next door.

3- 7245 E. Alondra Blvd., Paramount, California 90725

10- Again, I propped a socket to the back edge of the CPU card and wired the ACIA in at the appropriate address, using solder flux between the ACIA and the case for ease of removal of the card.

The second was the need for the mother board, which shouldn't have been a problem. I should have done it right the first time. I still have the scars from now where from Richard myself.

The last was the decoding of 8000h and 8080-8080h as off-board. That raised with the excellent help of Mr. Dave Gillett of Motorola Semiconductor Sales, over the phone.

The system now does, among a large number of other and such, several things that have earned it (and me) my wife's undying praise and support (we say "the future"). It printed our Christmas card addresses for us last year and

will continue to do so. It maintains our household accounts for us even balancing the checkbook. For that she will give me anything! It also gives her something to give me for Christmas.

I don't want to sound too commercial, but I cannot recommend PERCON too highly. They have worked, been patiently questioned, chucked, and perhaps cursed a time or two with me as the "unwashed" of my system made themselves known. Special thanks go to Mr. Jim Stufman, who has been infinitely patient and kind. Ditto to Mr. and Mrs. as you can see, and my 6800-MEK-D2-SS-50-LFD-400 seems to be doing quite well. I don't regret it a bit.

If I can be of any assistance to anyone with a similar problem, please feel free to write. I am no expert, but having seemed to talk with who has "been there" has been a real boon to me and I will be glad to try.

December 20, 1979

3406 Notre Dame Street  
 Seattle, WA 98103

Don Williams  
 '68' Micro Journal  
 Mixson, TN 37343

Dear Don,

Here's my check for \$3.50 for issue 1. That will make my collection complete!

I was interested in the CFM File Lister in the January '80 issue for two reasons: 1) I'd been meaning to put together that program, and 2) the comparison between using a PIA and an ACIA is worth a look. I'm enclosing my version for use with the ACIA and the SUTBUG monitor. Actual program length is about one-third the length of the PIA version. Interesting!

Santa Claus brought me a copy of Tiny Assembler 6800 by Jack Emerson and the listing program gave me a chance to try it out. The other attachment is a copy of the output of the disassembled version using a slightly modified version of the disassembler by Bob Lentz in the May '79 Byte. The ser and lch connection of the last byte was operator error in running the assembled version through an editor to make some other corrections without the disassembler. I did not have noticed the error. It's good to have both tools.

I'd like to add my commendation regarding the JPC product. I've been using the TC-3 high speed cassette interface for a good many months now, and the file manager CFM-3 for a few months. Both products are great, work as promised, and the documentation is outstanding! I've recently built a small module with a 2716 and a 7420 to replace the SUTBUG chip so that I could easily "boot" in the CFM-3 manager instead of going through several switch-flipping steps. The space in SUTBUG for the disc boot "D" command was used for a new "T" command since I don't use disc and probably won't for a long time because of the ease of use and low cost of the cassette mode of operation.

I look forward to the '68' Micro each month; keep up the good work.

Sincerely,

Ches Looney  
 Ches Looney

```

LOCN B1 B2 B3
0000      )*****
0000      )> CFM-3 UTILITY TO PRINT DIRECTORY
0000      )> MODIFIED FROM PROGRAM 111 JAN 80
0000      )> '68' MICRO BY MIDDAGH; PAGE 26
0000      )*****
0000      )
0000      )> DEFINITIONS:
0000      )
0000      )>NAMELG EQU  #AFBC  CFM NAME SPECIFIED FLAG
0000      )>TYPE EQU  #AF8B  CFM FILE TYPE CHECK FLAG
0000      )>NAMEA EQU  #AEDC  CFM DIRECTORY HEADER STRING
0000      )>GETID EQU  #AFAF  CFM SUBROUTINE TO GET ID SEGMENT
0000      )>MOTOFF EQU  #AA16  CFM SUBROUTINE TO TURN OFF RECORDER
0000      )>FBEGA EQU  #AFA7  CFM FILE BEGIN ADDRESS
0000      )>NAME EQU  #AFA1  CFM FILE NAME STORE
0000      )>TYPEMSG EQU  #AF26  CFM TYPE MESSAGE STRING
0000      )>FTYPE EQU  #AFA0  CFM FILE TYPE STORE
0000      )>CLFSP EQU  #AEEF  CFM CR LF SP STRING
0000      )>PORTAI EQU  #E07E  SUTBUG STRING PRINT SUBROUTINE
0000      )>OUT4MS EQU  #E0C8  SUTBUG 4 HEX CHAR PRINT SUBROUTINE
0000      )>OUTCH EQU  #E1D1  SUTBUG ASCII CHAR PRINT SUBROUTINE
0000      )
0000      )> PROGRAM SECTION:
0000      )
0000      )> ORG  #A7B0
0000      )> A7B0 7F AF BC >INIT CLR NAMELG
0000      )> A7B3 7F AF 8B > CLR TYPE
0000      )> A7B6 CE FE DC > LDX #NAMEA
0000      )> A7B9 80 E0 7E > JSR PORTAI
0000      )> A7BC 80 AF FC > JSR GETID
0000      )> A7BF 80 AA 16 > JSR MOTOFF
0000      )> A7C2 CE AF A7 > LDX #FBEGA
0000      )> A7C5 80 E0 C8 > JSR OUT4MS
0000      )> A7C8 80 E0 C8 > JSR OUT4MS
0000      )> A7CB 80 E0 C8 > JSR OUT4MS
0000      )> A7CE C6 06 > LDAB B6
0000      )> A7D0 CE AF A1 > LDX #NAME
0000      )> A7D3 A6 00 >NAME LDRB B,K
0000      )> A7D6 80 E1 D1 > JSR OUTCH
0000      )> A7D9 00 > INX
0000      )> A7DB 00 > DECB
0000      )> A7DE 26 F7 > BNE NAME
0000      )> A7E0 CE AF 26 > LDX #TYPEMSG
0000      )> A7E3 80 80 7E > JSR PORTAI
0000      )> A7E6 86 AF A6 > LDAB #TYPE
0000      )> A7E9 88 30 > ADDA #B30
0000      )> A7F3 80 E1 D1 > JSR OUTCH

```



```

A73A CE RE FE > LDX @R1FSP
A73D BD E0 7E > JSR PDTRA1
A740 20 CA > BRR PDTR
A740 > END

```

\*\*\* UNRESOLVED:  
\*\*\* SYMBOLS?

```

V
CRUP REFE FBRA AFA7 FME AFAL FIVE AFAD GETO AFAG
INIT A700 MOTF AR16 NAME AFBC OUTH E101 OUTS E0 8
PDRI 007E PDTR A7BC PHER REOC PWHM A723 TYPE AF8B
TYPE AF26

```

```

A700 7F AF8C CLP AF8C
A703 7F AF8B CLR AF8B
A706 CE REOC LHM REOC
A709 BD E07E JSR E07E
A70C BD AF8C JSR AF8C
A70F BD AR16 JSR AR16
A712 CE AFA7 LDX AFA7
A715 BD E0C8 JSR E0C8
A718 BD E0C8 JSR E0C8
A71B BD E0C8 JSR E0C8
A71E C6 06 LHM 6 06
A720 CE AFA1 LHM AFA1
A723 A6 00 JSR E101
A726 08 JIC
A729 26 F7 BNE A729
A72C CE AF26 LDM AF26
A72F BD E07E JSR E07E
A732 B6 AFA0 LDM AFA0
A735 88 20 ADC A 20
A737 BD E101 JSR E101
A73A CE REFE LDX REFE
A73D BD E07E JSR E07E
A740 20 CA BRR A7BC
A742 8

```

```

74EB A6 01 LDA A 10X
74ED 81 1F CNP A #31F
74EF 22 BC BHI CHIN
74F1 8D 02 BSR RUBI
74F3 20 88 BRR CHIN

```

```

74F5 86 08 RUBI LDA A #95 LOAD A BACKSPACE
74F7 8D 16 BSR OUTCHI
74F9 86 20 LDA A #80 LOAD A SPACE
74FB 8D 12 BSR OUTCHI
74FD 86 08 LDA A #85 BACK UP CURSOR AGAIN
74FF 20 0E BRR OUTCHI

```

```

7501 81 1F CHRPAT CNP A #31F CHECK FOR CONT CHAR
7503 22 08 BHI ASCII BRANCH IF PRINTABLE ASCII
7505 88 40 ADD A #40 CONVERT CONT CHAR TO PRINTABLE
7507 36 PSH A SAVE IT
7508 86 5E LDA A #1 LOAD UP ARROW
750A 8D 03 BSR OUTCHI PRINT IT
750C 32 PUL A GET A BACK
750D 01 ASCII NOP FILL REMAINING SPACE
750E 01 NOP

```

```

75 7 7E 12 86 OUTCHI JMP OUTCH

```

\* SET RETURN TO SMARTBUO JUMP IN DOS

```

728C ORG 728C
728C 7E E0 E3 JMP 728C JUMP TO SMARTBUO

```

END

NO ERROR(S) DETECTED

SYMBOL TABLE:

ASCII	750D	BACK	74E4	BREAK	0060	BRI	74A1
BS	0008	BUFFER	6FFF	BUFL	740A	CHIN	74AD
CHRPAT	7501	ECHOPL	A00B	ETCHR	74C8	INCR	72B9
LININ	74A6	OUTCH	7286	OUTCHI	750F	RUBI	74F7
START	74A6	XTMP	72FC				

NAME BACKSPACE PATCH FOR SMARTBUO/DOS568

PATCH 11 DOS568 WITH SMARTBUO BACKSPACE

ADAPTED FROM DAN JOHNSON'S SCOPE  
IN SEPT '79 OF '68 MICRO JOURNAL'

REVISED FOR SMARTBUO 9-79 ALSMITH

ASSEMBLED PATCH SHOULD BE APPENDED TO  
DOS568.31 OR DOS568.42 IN PLACE OF 'SBP'  
REF (SSB DISC MANUAL 2.8)

VALID KEYBOARD BACKSPACE COMMANDS:

CONTROL M  
CONTROL O

VALID BREAK COMMANDS:

CONTROL X  
ASCII GRAVE ACCENT

```

74A6 LININ EQU 74A6
72B9 OUTCH EQU 72B9
72B9 INCH EQU 72B9
6FFF BUFFER EQU 6FFF
72FC XTMP EQU 72FC
0008 BS EQU 008
0060 BREAK EQU 060
A00B ECHOPL EQU A00B
74A1 BRR EQU 74A1

```

10F IF TERM BS IS CONT O  
GRAVE ACCENT (BREAK KEY)  
SMARTBUO ECHO FLAG

```

74A6 74A6 CE 67 FF START ORG LDM #BUFFER
74A9 7F 72 FC STX XTMP
74AC 5F CLR B

```

```

74AD 80 19 CHIN BSR GETCHR INPUT CHARACTER FROM TERMINAL
74AF 81 0F CNP A #30F LOOK FOR A CONTROL 'O'
74B1 27 31 BEQ BACK IF IT IS BACKUP CURSOR
74B3 81 08 CNP A #85 LOOK FOR A CONTROL 'H'
74B5 87 2D BEQ BACK BACK UP ON THIS ONE TOO
74B7 81 60 CNP A #BREAK BREAK IF A '9' GRAVE
74B9 27 36 BEQ BRR1 ALSO BREAK IF A CONTROL X
74BB 81 18 CNP A #318 PUT A IN INPUT BUFFER
74BD 27 E9 BEQ BRR1 CHECK FOR A CR
74BF A7 01 STA A 10X IF NOT CHECK IF BUFFER IS FULL
74C1 81 0D CNP A #30D RETURN BACK TO THE INARDS OF DOS
74C3 26 15 BNE BUFL SAVE B
74C5 7E 73 53 JMP 67353 SAVE ECHO FLAG STATUS
74C8 37 GETCHR PSH B TURN ECHO OFF
74C9 FA A0 0B LDA B ECHOPL TURN ECHO ON
74CC 7F A0 0B CLR ECHOPL GET A CHARACTER FROM KEY BOARD
74CF 7C A0 0B INC ECHOPL RESTORE FLAG TO ORIGINAL STATUS
74D2 BD 72 89 JSR INCH
74D5 7F A0 0B STA B
74D8 33 PUL B RESTORE B
74 9 39 RTS

```

```

74DA C1 47 BUFL CNP B #347 IS BUFFER FULL?
74DC 27 CF BEQ CHIN IF YES BACK TO 00
74DE 8D 21 BSR CHRPAT BUMP CHAR COUNT
74E0 5C INC B BUMP BUFFER POINTER
74E1 08 INX BACK TO INPUT
74E2 20 C9 BRR CHIN

```

```

74E4 5D BACK TST B BACK TO TOP OF BUFFER?
74E5 27 C6 BEQ CHIN IF YES GO GET ANOTHER CHAR
74E7 8D 0C BSR RUBI BACKUP CURSOR AND ERASE CHAR
74E9 5A D C B BACKUP CHAR COUNT
74EA 09 DSX LIKEWISE BUFFER POINTER

```

0001 REM THIS PROGRAM WILL ALLOW YOU TO  
READ FROM 1 DISK AND WRITE TO THE OTHER.  
IT IS WRITTEN IN COMPUTERWAKE BASIC.  
0002 REM JR JOHNSON  
0003 REM JOHNSON MICROCOMPUTER  
0004 REM LAS VEGAS NEV.  
0005 REM WORKS WITH TSO MINIFLEX  
0006 POWER (28812.1)  
0007 OPEN #1,DATA  
0008 SCRATCH #1  
0009 POWER (28812.6)  
0010 OPEN #2,DATA: REM FILE HAS TO BE ON  
DISK 0  
0011 REM #2,AL,04,0,0,0  
0012 IF STATUS #2-2 CONT 100  
0013 WRITE #1,AL,04,0,0,0  
0014 GOTO 70  
0015 CLOSE #1,#2

NAME LOGS

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37

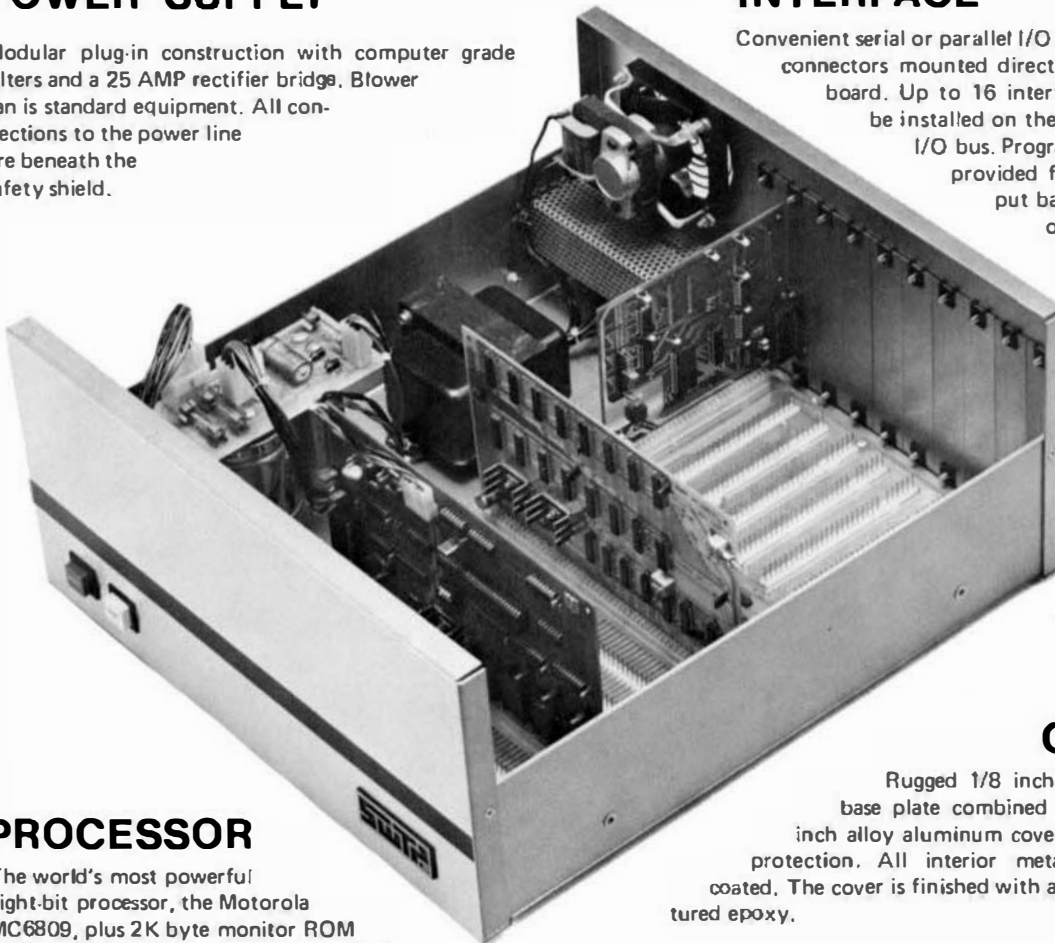
# WE HAVE A 6809 FOR YOU

## POWER SUPPLY

Modular plug-in construction with computer grade filters and a 25 AMP rectifier bridge. Blower fan is standard equipment. All connections to the power line are beneath the safety shield.

## INTERFACE

Convenient serial or parallel I/O cards have DB-25 connectors mounted directly on the circuit board. Up to 16 interface devices may be installed on the address decoded I/O bus. Programming strips are provided for input and output baud rate selection on each port. All outputs are fully buffered.



## PROCESSOR

The world's most powerful eight-bit processor, the Motorola MC6809, plus 2K byte monitor ROM that is 2716 EPROM compatible and full buffering on all output lines. Built-in multiuser capability, just add I/O cards to operate a multi-terminal system.

## CABINET

Rugged 1/8 inch alloy aluminum base plate combined with a solid 1/8 inch alloy aluminum cover for unsurpassed protection. All interior metal is conversion coated. The cover is finished with a super tough textured epoxy.

**MEMORY**— You can purchase the computer with either 8K bytes of RAM memory (expandable to 56K), or with the full 56K. The efficient, cool running dynamic memory used in this system is designed and manufactured for us by "Motorola Memory Systems Inc."

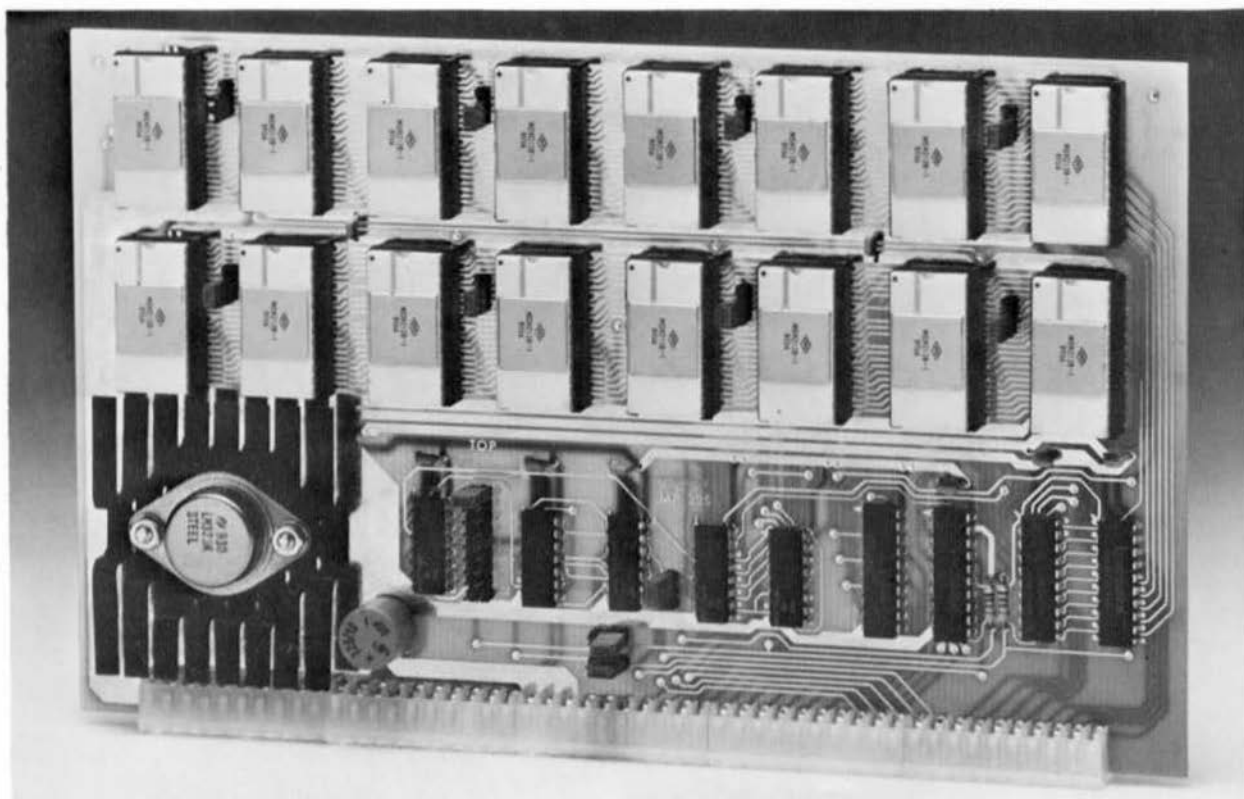
**PERIPHERALS**— The wide range of peripheral hardware that is supported by the 6809 includes: dot matrix printers (both 80 and 132 column), IBM Electronic 50 typewriter, daisy wheel printers, 5-inch floppy disk system, 8-inch floppy disk systems and a 16 megabyte hard disk.

**SOFTWARE**— The amount of software support available for the 6809 is incredible when you consider that it was first introduced in June, 1979. In addition to the FLEX9 operating system, we have a Text Editor, Mnemonic Assembler, Debug, Sort-Merge, BASIC, Extended BASIC, MultiUser BASIC, FORTRAN, PASCAL and PILOT.

69/K Computer Kit with 8K bytes of memory . . . . .	\$ 495.00
69/A Assembled Computer with 8K bytes of memory . . . . .	\$ 595.00
69/56 Assembled Computer with 56K bytes of memory. . . . .	\$1,495.00



SOUTHWEST TECHNICAL PRODUCTS CORPORATION  
219 W. RHAPSODY  
SAN ANTONIO, TEXAS 78216 (512) 344-0241



## UNIVERSAL static memory card

- ★ 32K bytes-ROM, RAM, EPROM or a combination
- ★ SS-50 A&C compatible with 16 and 20 bit address decoding
- ★ Compatible with all SWTPC 6800 and 6809 computers

This is the most versatile memory card you can buy. Our S-32 may be populated with up to 32K of static RAM, EPROM, or ROM, or any 4K block combination of these that you may desire. Any 5-volt 2716 pinout compatible memory may be used in this card. Any 4K block of the memory may be jumper block programmed for RAM or ROM use. This feature makes this the ideal memory for those process control applications that require a mixture of ROM and RAM memory. The board is fully compatible with all SWTPC 6800 and 6809 computers.

The power requirement for the board is only 1.75 amps at 5.0 volts with a full 32K of RAM installed.

### S-32 Circuit card assembled

—less memory IC's (uses up to 16). \$99.50

2716 Type EPROM for above . . . . . \$50.00 ea.

16K (2K x 8) Static RAM for above  
(4016 or 2128). . . . . \$50.00 ea.



SOUTHWEST TECHNICAL PRODUCTS CORPORATION  
219 W. RHAPSODY  
SAN ANTONIO, TEXAS 78216 (512) 344-0241



'68' Micro Journal

TYPE	FOR	RESTART					
			6800			6809	
C000	RESTART	LIX		\$RESTART-\$80	LOI		\$RESTART-\$80
		LIX		\$RESTART	LOI		\$RESTART
		LDA		\$80,X	LDA		,X,
		CHP	A	\$80,X	CHPA		,X,
		BEQ	A	\$6800P	BEQ		\$6800P
		STA	A	\$CPUYPE	STA		\$CPUYPE
		JMP		\$6809	JMP		\$6809
		CLR		\$CPUYPE	CLR		\$CPUYPE
	66800P	JMP		\$6800	JMP		\$6800P

The two sets of mnemonics generate the same object code (strange as it looks), which will execute as follows in the 2 CPU's. First (because it was) the 6800 loads the index register with \$8F80, then it stack pointer with \$0000 (a don't care in this case), then A with the contents of \$0000 (\$800 & \$80). The next first instruction is a branch if less than, which will branch to \$0000 with the contents of \$0000 AND increments X. Next both CPU's compare A with either the location loaded from (the 6800) or the location loaded from +1 (the 6801). Of course, only the 6800 gets a match so it takes the branch and jumps to \$0000. The fall through code is shared for reference by the 6801 and 6802, but not for use in a like report.

Using these methods, I am currently developing a multi part monitor for my system which, in its smaller version, includes memory examination and change in 3 formats: multiple breakpoint points, single stopping, I/O, load/punch memory, search for a byte of data, examine and change registers, etc. When the second part is added, an additional search/examine/change socket is added, the search function expands up to a 16 byte seek and 16 byte target in either HEX or ASCII and punching and loading of tape in replaceable format is also added. The next part adds the option of examining and changing memory in endocore format.

Especially,

Derek Gibson  
Barwick Systems

Attestado en

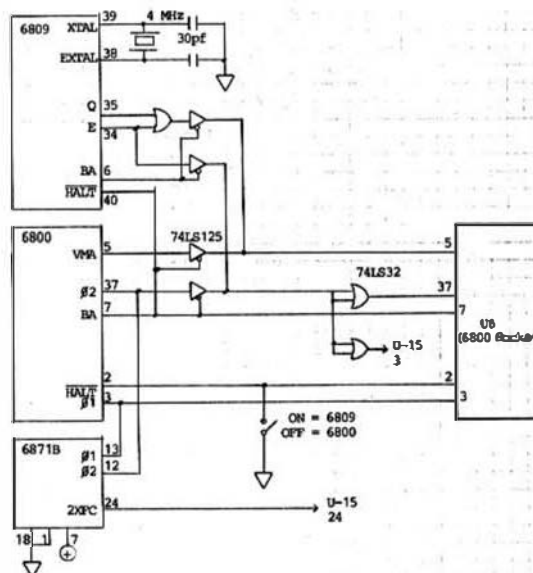


TABLE 1

Circuit to allow use of either the 6800 or 6809 in a Motorola 6800 MEX D2 evaluation kit. The rest of the connections are shown in the wire list.

'68' Micro Journal  
301B Hambl Rd.  
P.O.Box 849  
Hixson, Tennessee 37343

Dear Sir,

Please find enclosed information on modifying Dr. Chuck Adams' Line Editor. I worked out the additional Print command. The command was modified from Peter Stark's program in the Jan '79 issue of Kilobaud. I submit this information for your use either as an article or a letter-to-the-editor. If the material is inappropriate please return it in the SASE.

At any rate, I enjoyed Chuck's editor and the challenge of working out the modification for my own use. All those disk patches are neat but keep up the work finding machine language programs of a useful nature.

Having mentioned that other magazine let me say that as it gets bigger the less interesting it becomes to me. I think '68' Micro is terrific. Keep it up.

Respectfully yours,

Bruce M. Henry  
St Rte 3 Box 170  
Farmington, MN 57401  
505-632-2165

### LINE EDITOR MODIFICATIONS

by

Bruce N. Henry

Star Route 3 Box 170  
Farmington, New Mexico 87401

This is a modification of Chuck Adams' program LINE EDITOR listed in this journal in October, 1979, page 23. I liked his program and learned some new tricks from studying it. In spite of it being a great program, I needed to modify it.

One reason is because I'm still using Mikbug. With Mikbug I couldn't get out of the text Print routine by using a CTRL-C. So, one modification is to print a set of lines. Now I can print from one line number to another and stop printing. The format for this additional command is: P line#, line#. If both line numbers are the same only that line is printed. Because of the nature of the algorithm the largest line number cannot be used. The command works best when

Continued on page - 36

FUNCTION	06	6800	6809
GROWN	1	1	1
WALT	2	FIG. 1	-
D1	3	FIG. 1	-
WQ	4	4	3
WMA	5	FIG. 1	(5 W.C.)
WMT	6	6	2
BA	7	FIG. 1	FIG. 1
VOC (+S)	8	8	4, 2, 3, 3, 3, 6
A0	9	9	8
A1	10	10	9
A2	11	11	10
A3	12	12	11
A4	13	13	12
A5	14	14	13
A6	15	15	14
A7	16	16	15
B	17	17	16
A9	18	18	17
A10	19	19	18
A11	20	20	19
GRD	21	21	-
A12	22	22	20
A13	23	23	21
A14	24	24	22
A15	25	25	23
D7	26	26	24
D6	27	27	25
D5	28	28	26
D4	29	29	27
D3	30	30	28
D2	31	31	29
D1	32	32	30
D0	33	33	31
B/W	34	34	32
W.C.	35	-	-
D08	36	36	-
D2	37	FIG. 1	-
W.C.	38	-	-
TBC	39	39	-
TBC	40	40	37

Wire list for connecting a 6800 or 6809 into a Motorola 6800 XER 02. Chips are connected as indicated here and in Figure 1.

## BOOKEEPING Final part?

Please note the letter following  
As errors or omissions are discovered, please let us know, the changes will appear as we receive them. Those who have received disk with this error will receive a copy of the amended disk as received from the author. All that is required is the return of the original disk and a updated version will be forwarded, postpaid by 68 Micro Journal.

Due to a typo error the original cost was to have been \$16.50 for a 5 1/4" disk with the programs as submitted by the author. It was published as \$6.50. We honored the 'error' price, but for all orders received after April 30, 1980 the \$16.50 price will be required. There is just no way we can furnish the disk, make the copies and pay the postage for \$6.50. We try hard to keep all cost for our readers at the lowest possible price for everything, but I trust that you can understand this increase.

DMW

William Stock  
1125 Lols Dr.  
Cincinnati, OH 45237

```
0001 REM BUDGET PROGRAM ICLEVER).
0002 REM BUDGET.RAS
0010 GOTO 1000
0091 REM
0093 REM SUBROUTINES
0095 REM
0097 REM CLEAR SCREEN
0098 REM
0100 PRINT CHR$(16);CHR$(22);CHR$(0);CHR$(0);CHR$(0);
0110 RETURN
0190 REM
0199 REM FREEZE SINGLE ACCT
0200 A(5,X)=A(5,X)+A(4,X)
0210 A(4,X)=0
0220 A(2,X)=INT(A(4,X)*100/M2+.5)/100
0230 RETURN
0290 REM
0299 REM FREEZE ALL ACCTS
0300 FOR X=1 TO M1
0310 A(5,X)=A(5,X)+A(4,X)
0320 A(4,X)=0
0330 NEXT X
0340 RETURN
0390 REM THAW SINGLE ACCT
0400 A=VAL(A$)
0410 FOR Y=1 TO N1
0420 IF A(1,Y)=B THEN 460
0430 NEXT Y
0440 PRINT "NOT FOUND"
0450 GOTO 490
0460 A(4,Y)=A(4,Y)+A(5,Y)
0470 A(5,Y)=0
0480 A(2,Y)=INT(A(4,Y)*100/M2+.5)/100
0490 RETURN
0499 REM THAW ALL ACCTS
0500 FOR Y=1 TO N1
0510 A(4,Y)=A(4,Y)+A(5,Y)
0520 A(5,Y)=0
0530 A(2,Y)=INT(A(4,Y)*100/M2+.5)/100
0540 NEXT Y
0550 RETURN
0991 REM
0993 REM PROGRAM STARTS HERE
0995 REM
1000 GOSUB 100
1010 OPEN #1,O,PRR
1020 READ #1,U1,V2,V3,V4,V5,U6,V7
```

Gentlemen:

I have just received a phone call from one of our fans, and he informed me that the Book Base pack I sent you has an error: The contents of the program named MENU is really START!

The enclosed diskette is corrected. Sorry for the inconvenience, and I hope you can get the listings corrected before publication - or at least include a note.

Sincerely,

William R. Stock

```
1125 REM PUT ENTIRE EXPENSE SECTION IN MEMORY
1127 REM
1130 FOR X=1 TO M1
1140 READ B1,A11,X1,A(4,X),A(5,X),G1
1150 IF A(1,X)<=V6 THEN 1140
1155 A(4,X)=A(4,X)+A(4,X)
1160 NEXT X
1170 RESTORE #1
1180 PRINT "ENTER BUDGET PERIOD"
1190 PRINT " 1 = ANNUAL"
1200 PRINT " 12 = MONTHLY"
1210 PRINT " 24 = SEMI-MONTHLY"
1220 PRINT " 26 = BI-WEEKLY"
1230 INPUT " 32 = WEEKLY " ,M2
1240 M2=ABS(M2)
1245 INPUT "ANNUAL INCOME",I
1247 GOTO 1390
1250 GOSUB 100:REM INPUT ROUTINE
1251 F=1
1260 FOR X=1 TO M1
1270 IF A(5,X)<>0 THEN 1300:REM ACCT FROZEN
1280 PRINT A(5,X);TAB(10);A(2,X);TAB(25);A(3,X)
1290 INPUT #5,M5
1300 IF M5="T" THEN 1400:REM THAW THIS ACCT
1310 IF M5="F" THEN 1450:REM FREEZE THIS ACCT
1320 IF ASC(M5)>0 THEN A(2,X)=VAL(M5):A(4,X)=A(2,X)*M2
1330 IF M5="F" THEN 1450:REM FREEZE ENTERED AMT
1340 IF ASC(M5)=0 THEN M3=M2/100:1340
1350 M3=VAL(M3)
1360 A(4,X)=A(2,X)*M3
1370 A(2,X)=INT(A(4,X)*100/M2+.5)/100
1380 NEXT X
1390 I1=0:I2=0:REM THAWED AND FROZEN TOTALS
1400 FOR X=1 TO M1
1410 I1=I1+A(4,X):I2=I2+A(5,X)
1420 NEXT X
1425 IF F=0 THEN 1450
1430 IF I1+I2<=I THEN 1700:REM UNDER INCOME
1440 PRINT "OVER INCOME BY ";I1+I2-I
1450 IF I1=0 THEN GOSUB 500:I1=I2:I2=0
1460 F1=100*(I-I2)/I1
1465 REM CALCULATE SUGGESTIONS
1470 FOR X=1 TO M1
1480 A(3,X)=INT(A(4,X)+I1/M2+.5)/100
1490 NEXT X
1500 GOTO 1250:REM PLAY IT AGAIN, SAM
1600 GOSUB 400:REM THAW THIS ACCT
1610 GOTO 1390:REM RECALCULATE
1650 GOSUB 200:REM FREEZE THIS ACCT
1660 GOTO 1300:REM NEXT
1699 REM PRINT BUDGET
1700 GOSUB 100
1710 PRINT "ACCT DESCRIPTION";TAB(25);"AMOUNT"
1720 PRINT
1730 GOSUB 300:REM FREEZE ALL ACCTS
1740 FOR X=1 TO M1
1750 PRINT A(1,X);A(5,X);
1760 A=INT(A(5,X)*100/M2+.5)/100
1770 AB=STR$(A)
1780 IF A=INT(A) THEN AB=AB+".0"
1790 IF A=10*INT(A/10) THEN AB=AB+".00"
1800 PRINT TAB(32-LEN(AB));AB
1810 NEXT X
1820 PRINT
1830 IF I-I1-I2>=1 THEN PRINT "9";I-I1-I2;"EXTRA THIS YEAR"
1840 IF I-I1-I2<1 THEN PRINT "(3-I1-I2)*100;"CENTS EXTRA THIS YEAR"
1850 INPUT "DO YOU WANT TO RE-DO IT",A9
1860 IF A9="" THEN 1850
1870 IF LEFT$(A9,1)="Y" THEN I2=I2+I1:I1=0:GOTO 1450
1880 IF LEFT$(A9,1)(">"M" THEN 1850
1890 GOSUB 100
1900 PRINT "PUT OLDEST G/L ON 80"
1910 INPUT "IS IT THERE",A9
1920 IF A9="" THEN 1910
1930 IF LEFT$(A9,1)(">"Y" THEN 1900
1950 OPEN #2,O,GLMASTER
1960 SCRATCH #2
1970 FOR X=1 TO #1+1
1980 READ #1,G1,G2,G4
1990 IF EOF(1)=1 THEN 2090
2000 WRITE #2,G1,G2,G4,A(5,X)
2010 IF G4>V7 THEN 1980
2020 IF G1<=V6 THEN 1980
2030 IF G1<A(1,X) THEN 2050:REM NOT IN SYNC
2040 NEXT X
```



```

2050 GOSUB 100
2060 CLOSE #1,#2
2070 PRINT "DISASTER; RE-BOOT!"
2080 RUN
2090 CLOSE #1,#2
2100 OPEN #1,OLNIST
2110 OPEN #2,OLMIST
2120 SCRATCH #2
2130 READ #1,A,B,C,D
2140 IF EOF(1)=1 THEN 2170
2150 WRITE #2,A,B,C,D
2160 GOTO 2130
2170 CLOSE #1,#2
2180 DOSUB 100
2190 PRINT "CURRENT 6/L ON 00"
2200 PRINT :PRINT "I NEED SYSTEM DISC ON 80"
2210 INPUT "IS IT THERE",A$
2220 IF A$="" THEN 2210
2230 IF LEFT$(A$,1)<>"Y" THEN 2200
2240 CHAIN 0.MENU

```

```

0001 REM PRINTS BUDGET.
0002 REM ACCT DESCR; 1: CURRENT 5; BUDGETED 5
0003 REM ON STAGGERED DUES DUE
0004 REM TO 32 CHARACTER WIDTH OF 1024 SCREEN.
0005 REM B.PNT.BAS
0010 GOTO 1000
0100 PRINT CHR$(16);CHR$(22);CHR$(10);CHR$(0);CHR$(0);
0110 RETURN
0200 OPEN #1,0.PRM
0210 READ #1,A,V2,V3,V4,V5,V6,V7,V8,V9
0220 CLOSE #1
0230 J1=INT(A/10000)
0240 A=INT(A/100)-J1*100
0250 ON J1 DOSUB 270,280,290,300,310,320,330,340,350,360,370,380
0260 A=A/365
0270 RETURN
0280 A=A+31:RETURN
0290 A=A+59:RETURN
0300 A=A+90:RETURN
0310 A=A+120:RETURN
0320 A=A+151:RETURN
0330 A=A+181:RETURN
0340 A=A+212:RETURN
0350 A=A+243:RETURN
0360 A=A+273:RETURN
0370 A=A+304:RETURN
0380 A=A+334:RETURN
1000 GOSUB 100:GOSUB 200
1010 PRINT "BUDGET REPORT"
1020 PRINT "DESCR;TAB(19);CUR SUB"
1040 PRINT :A$=""
1050 OPEN #1,OLMASTER
1060 READ #1,B1,B2,B3,B4
1070 IF 0<V6 THEN 1060
1075 IF 0<V7 THEN 1200
1080 B=6+21*(B3-S1R1-B2)
1090 IF B3=0 THEN V6="00";B=0:GOTO 1130
1100 B=B3+A
1110 V6=STR$(-INT(B2+100/B+.5))
1120 B=INT(B+100+.5)/100
1130 B1=B1+B
1135 IF B2=0 THEN 1060
1140 B5=S1R1(B)
1150 D5=B5-A5
1160 D5=LEFT$(D5,16)
1170 PRINT D5;TAB(26-LEN(C5));C5
1180 PRINT TAB(16-LEN(V6));V6;TAB(32-LEN(B5));B5
1190 PRINT :GOTO 1060
1200 CLOSE #1
1210 C5=STR$(-G)
1220 B=0
1230 B5=STR$(B)
1240 IF B=0 THEN V6="00";GOTO 1260
1250 V8=B1R1(-INT(G+100/B+.5))
1260 PRINT
1270 PRINT "TOTAL";TAB(26-LEN(C5));C5
1280 PRINT TAB(16-LEN(V6));V6;TAB(32-LEN(B5));B5
1290 PRINT
1300 INPUT "RETURN. ",A$
1310 CHAIN 0.MENU

```

```

0001 REM READS A/P AND ADDS AMT DUE FROM LAST UPDATE
0002 REM TO DATE ENTERED.
0003 REM IF DATE DUE = 0 IT IGNORES THE ACCT.
0004 REM IF AMT DUE = 0 IT USES THE BALANCE.
0005 REM CASH.BAS
0010 GOTO 1000
0020 REM
0030 REM SUBROUTINES
0040 REM
0100 PRINT CHR$(16);CHR$(22);CHR$(0);CHR$(0);CHR$(0);
0110 RETURN
0120 REM
0130 REM JULIAN DATE CONVERT
0140 REM
0200 J1=INT(B/100)
0210 J2=B-J1*100
0220 ON J1 DOSUB 240,250,260,270,280,290,300,310,320,330,340,350

```

```

0230 RETURN
0240 RETURN
0250 J=J+31:RETURN
0260 J=J+59:RETURN
0270 J=J+90:RETURN
0280 J=J+120:RETURN
0290 J=J+151:RETURN
0300 J=J+181:RETURN
0310 J=J+212:RETURN
0320 J=J+243:RETURN
0330 J=J+273:RETURN
0340 J=J+304:RETURN
0350 J=J+334:RETURN
0400 IF P3=INT(P3) THEN P5=P5+1:B"
0410 IF P3=10+INT(P3=10) THEN P5=P5+1:B"
0420 RETURN
0993 REM
0995 REM START PROGRAM HERE
0997 REM
1000 OPEN #1,0.PRM
1010 READ #1,V1
1020 CLOSE #1
1030 B=INT(V1/100)
1040 DOSUB 200
1050 B1=J
1060 GOSUB 100
1070 INPUT "DO YOU WANT DETAIL",B1
1080 IF B1="" THEN 1070
1090 D5=LEFT$(D5,1)
1100 IF D5<>"Y" THEN IF D5<>"N" THEN 1070
1110 INPUT "CASH NEEDED TO HANDOUT",V2
1120 IF V2<101 THEN 1110
1130 IF V2>1231 THEN 1110
1140 B=V2
1150 GOSUB 200
1160 D2=J
1170 IF D2<D1 THEN D2=D2+365
1180 INPUT "DATA OK",A1
1190 IF A1="" THEN 1180
1200 A1=LEFT$(A1,1)
1210 IF A1<>"Y" THEN 1060
1220 GOSUB 100
1230 PRINT "CASH REQUIRED FROM"
1240 PRINT TAB(31;V1);TO ";V2
1250 A=INT(V2/100-1)+100
1255 IF M=0 THEN M=1200
1260 OPEN #1,1.APMMASTER
1270 READ #1,P1,P2,P3,P4
1275 IF EOF(1)=1 THEN 1500
1280 IF P4=0 THEN 1270
1290 IF P3=0 THEN P3=P2:REM NO PAY DUE, USE BAL
1300 IF P3=0 THEN 1270
1310 B=P4:IF B<100 THEN B=B+M:REM DB FORMAT NEEDS MONTH ADDED
1320 GOSUB 200
1330 IF J<D1 THEN J=J+365:REM POSSIBLE YEAR ROLLOVER
1340 IF J>D1 THEN IF J<D2 THEN 1360
1342 B=P4:IF 0<100 THEN B=B+M+100:REM TRY NEXT MONTH
1343 IF B>1231 THEN B=B-1200
1344 DOSUB 200
1346 IF J<D1 THEN J=J+365:REM YEAR ROLLOVER, MAYBE
1348 IF J>D1 THEN IF J<D2 THEN 1360
1350 GOTO 1270
1360 C=C+P3:REM HLT
1370 IF D5="N" THEN 1270
1380 PRINT P1;P5;
1390 P5=STR$(-P3)
1400 GOSUB 400
1410 PRINT TAB(30-LEN(P5));P5
1420 GOTO 1270
1500 PRINT
1510 PRINT "TOTAL CASH REQUIRED:";
1520 P3=C+P5:STR$(P3)
1530 GOSUB 400
1540 PRINT TAB(30-LEN(P5));P5
1550 PRINT
1560 CLOSE #1
1570 INPUT "RETURN. ",A$
1580 CHAIN 0.MENU

```

#### BOOK REG. TXT

ASH,U=1,S=0  
GEI,0.BASIC.CMD  
0.BOOKS.DIN

#### BKSL JCL

COPY,0.JAPP.BAS,1  
COPY,0.JPNT.BAS,1  
COPY,0.TRAN.BAI,1.JOURNAL.DAT  
COPY,0.TRAN.BAT,1  
COPY,0.COPY.CMD,1  
0.BKSL.DUE  
COPY,0.GAER.BAS,1  
COPY,0.GSRI.BAS,1  
COPY,0.GPM1.BAS,1  
COPY,0.AATH.BAS,1  
COPY,0.TRAN.BAT,1.OLMASTER.BAI  
COPY,0.TRAN.BAT,1.OLNST.BAT

```

COPY,O.TRAN.DAT,1.GLTRAN.DAT
COPY,O.COPY.CMD,1
O.BK53.QUE
COPY,O.PUB.BAS,1
COPY,O.MAIN.BAS,1
COPY,O.TRAN.DAT,1.APST.DAT
COPY,O.IRAN.DAT,1.APRASTER.DAT
COPY,O.TRAN.DAT,1.APTRAM.DAT
COPY,O.COPY.CMD,1
O.BK54.QUE
O.BK51.QUE
EXEC,O.BK51.JCL
Books, JCL
COPY,O.BUDGET.BAS,1
COPY,O.BPNT.BAS,1
COPY,O.RECOV.BAS,1
COPY,O.START.BAS,1
COPY,O.PSRT.BAS,1
COPY,O.INSTALL.BAS,1
COPY,O.APD.BAS,1
COPY,O.PL.BAS,1
COPY,O.CASH.BAS,1
COPY,O.CHNG.BAS,1
COPY,O.APIN.BAS,1
COPY,O.MENU.BAS,1
COPY,O.BOOKS.BIN,1
COPY,O.BOOKS.BIN,1
O.BK51.QUE
EXEC,O.BK51.JCL

```

#### DIRECTORY OF DAIVE MUNDER 1

FILE#	NAME	TYPE	BEQIN	END	SIZE
2	START	.BAS	01-0A	02-03	12
3	MENU	.BAS	02-04	02-0F	12
4	APIN	.BAS	02-10	04-0A	31
5	PSRT	.BAS	04-0B	06-02	28
6	PUB	.BAS	06-03	06-0B	9
7	JAPP	.BAS	06-0C	06-12	7
8	BSRT	.BAS	07-01	08-09	27
9	GMER	.BAS	08-0A	09-04	13
10	JPNT	.BAS	09-05	0A-09	23
11	BPNT	.BAS	0A-0A	0C-09	36
12	PL	.BAS	0C-0A	0B-05	14
14	MAIN	.BAS	0E-07	0F-11	29
15	RECOV	.BAS	0F-12	10-0E	15
16	MMO	.BAS	10-0F	10-11	3
17	INSTALL	.BAS	10-12	11-0A	11
18	APD	.BAS	11-0B	13-04	30
19	BUDGET	.BAS	13-05	15-02	34
20	BPNT	.BAS	15-03	18-0F	13
30	CASH	.BAS	17-02	18-02	19

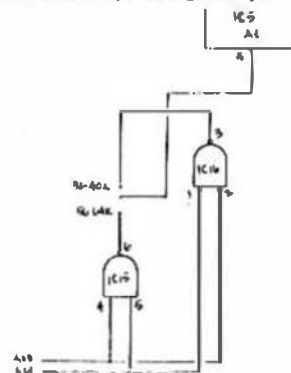
FILES=31, SECTIONS=405, LARGEST=34, FREE=207

#### DMAPI Controller Board Modification for Decoding at F000-F3FF

The following modifications will permit the controller board to be used in both 6800 and 6809 systems. The 32K-48K header is used to configure the applicable system. The 32K pin is used for 6800 and the 48K pin for 6809. It should be noted that the 48K pin has been changed to 56K.

Make all soldering changes on the bottom of the board using a light gauge wire.

- 1) From the bottom of the board, cut the land connecting IC15-4 and IC15-5.
- 2) Solder a wire from IC15-4 to IC16-2.
- 3) From the top of the board, cut the land connecting the 32K pin on the 32K-48K header and A14.
- 4) From the bottom of the board, cut the land connecting IC15-3 and IC5-5.
- 5) Solder a wire from IC16-3 to the 32K pin on the 32K-48K header.
- 6) From the top of the board, cut the land connecting the center pin on the 32K-48K header and IC16-1.
- 7) Solder a wire from IC16-1 to IC15-5.
- 8) Solder a wire from the center pin on the 32K-48K header to IC5-5.
- 9) Include these logic changes in your DMAPI documentation.



**SWTPC** Southwest Technical Products Corp  
219 W. Highway  
San Antonio, Texas 78216

February, 1980

#### NEWSLETTER #2

This week we will begin shipping version 2.6 of the FLEX 8 operating system with all disk drives. Version 2.6 is the final step in upgrading all of our 6809 systems to the new hardware and dual I/O arrangement first used in the "S" systems. With Ver. 2.6 it is no longer necessary to match the DOS to the system. The DOS now automatically checks the system, determines which type it is, and configures itself accordingly. This obviously makes life a lot simpler. There is no longer any question of, "Does he have the right version of DOS?" Version 2.6 of FLEX 8 runs on all current production 6809 systems.

You may have noticed that we have a number of utilities in the DOS package that you get with our drives that are not included in the version that is sold by TSC. All of these utilities and features have been added by SWTPC and any questions should be referred to us.

The PRINT utility has been replaced with a newer, more versatile Print Spooling Program. PSP supports speaking to slower, manuscript quality printers such as the SP-3 Quine printer and the IBM Electronic Model 50 typewriter in addition to most common serial and parallel printers. The Q utility has been enhanced to provide support for ten, twelve, or fifteen characters per inch printwidths, in either six or eight line per inch formats.

Version 2.6 now supports the MP-R PROM programmer for 6809 systems. This is done with READ, PROM and WRITEPROM in conjunction with FIX. FIX itself has been upgraded to handle binary files containing an arbitrary large number of blocks and the length limit on newly created blocks has been lifted.

Two utilities have been added that automatically answer Yes, or No to prompts for programs like DELETE and COPY. COPY itself has been enhanced to retain a file's protection attributes and last-altered date. A TOUCH utility has been provided to change the last-altered date of a file to the current date.

Both the mini floppy version of FLEX and the USEMF command have been upgraded to fully support the DC-3 controller used with the MF 68 double-sided disk drives. If you would like a copy of FLEX 8 Version 2.6 and a manual, they are available for \$35.00.

In response to many requests, we are making available our S/D9 chassis, power supply, master board and I/O available off-board a processor card, or memory. This will make it possible to put together systems with the new 32K static memories for special applications, or in situations where expansion is desired, but current needs are less than 128K. The price for the S/D9 (S/D9 computer less processor card and memory) is \$490.00. We are also offering the "S" system transformer for those who want a large power supply capacity in the 56K systems. Part number 352P9. Price is \$49.95.

Now the bad news. We have come to the point where it is necessary to increase prices on some items. Increases in the price of aluminum, copper, utilities, taxes, wages, etc. have become great enough to make this step necessary.

#### THE BIT BUCKET

Where all that 'good stuff' falls.  
Something for everyone.

February 20, 1980

'68 Micro Journal  
3018 Hamill Road  
Mixon, Tennessee

Done

Our local 68XX club has a couple of 6809-DMAPI systems up and running.

The SWTPC DMAPI modifications were unacceptable since the resulting controller board could not be used in a 6800 system. Our DMAPI boards have been modified in accordance with the attached instructions, and are being used in both 6800 and 6809 systems.

Sincerely,

Jack D. Johnson  
2816 Wood Creek Rd.  
Midwest City, Ok

# PERCOM ADDS 64K RAM CARD TO LINE OF EXORCISER® SYSTEM COMPATIBLE PRODUCTS

Orland, Texas - February 25, 1980 - Harold Mauch, president of Percom Data Company, today announced that the company is now manufacturing a 64K dynamic RAM micro-module for the Motorola EXORciser® and other EXORciser® bus compatible systems.

Percom manufactures mini-disk systems for the EXORciser®.

Designated the M64EX™, the Percom memory module features transparent refresh and optional parity check, yet sells for over \$1,000 less than the comparable Motorola 64K dynamic RAM micro-module®.

Mauch said the low price results from a design that incorporates state-of-the-art memory controller circuitry and address multiplexers. These advanced ICs permit cost-effective implementation of transparent refresh and other features.

The M64EX™ also includes an exclusive address translator circuit that accommodates program-controlled memory allocation, a feature useful in applications such as multi-tasking.

The M64EX™ permits any combination of 4K blocks of RAM -- within the upper 32K-bytes of memory space -- to be enabled or disabled with an on-board DIP switch. This enable-disable capability may be used with the address translator to implement functions such as write-protection of program-selected memory blocks.

## Other features include:

- \*DIP socket mounting of all RAM and complex IC chips.
- \*Three-state buffered interfacing with the system bus.
- \*An extensive capacitor bypass grid to minimize circuit-generated noise.
- \*Low power drain.
- \*0.062-inch PC board is double-sided plated-through FR4-010 epoxy glass. Printed wiring is 2 oz. copper with gold-over-nickel edge contacts and reflowed tin-lead plate conductors.

## Prices:

- M64EX™ w/64K-bytes RAM. . . . . \$ 875.00
- M64EX™ w/64K-bytes RAM & Parity Check . . . . . \$1,025.00
- M64EX™ w/48K-bytes RAM. . . . . \$ 795.00
- M64EX™ w/48K-bytes RAM & Parity Check . . . . . \$ 945.00

Prices are for assembled and tested modules. Optional burn-in charges and OEM quantity pricing schedule available on request.

Orders may be placed or additional literature requested through Percom's toll-free order number, 1-800-527-1592.

\* Trade mark of Motorola Corporation  
tm trademark of Percom Data Company, Inc.



**Intersystems**  
CORPORATION

Bldg 458  
Petersborough, N.H. 03458  
Phone (603) 924-6148

1980 March 15

Don Williams, Editor & Publisher  
"68 Micro Journal"  
1010 Hamill Rd  
Hixson Tennessee 37343

Dear Mr Williams:

The sample program below permits simple formatting of tabs and forward backward printing (if desired) using TSC BASIC or XBASIC. I imagine that your readers would like to know about this feature.

The program utilizes fielded variables in a buffer which is never used for file I/O; it is a "dummy buffer". In the following section, this buffer is allocated, and an array of fields, an array of bytes, and a byte string are overlay defined. The language used is TSC Precompiler Basic.

```
open "DUMMY.BUF" as 1
read fieldcnt: fieldcnt=fieldcnt-1: dim fields(fieldcnt)
bufsiz=0: for i=0 to fieldcnt: read fieldiz:
field i).bufiz as buffers, (fieldiz as field$(i)
bufsiz=bufsiz+fieldiz: next i: bufiz=bufsiz-1: dim bufbyte(bufiz)
```

```
for i=0 to bufiz: field $i,1 as buffers, 1 as bufbyte(i): next i
field $i, bufiz+1 as buffers: lset buffers=" "
data <fieldcnt>,<fieldiz0>,<fieldiz1>,...
open "<device driver file name>" as 0
reverse$=chr$(10)+chr$(27)+*6 *: reverse=0
```

In the above data statement the terms in angle brackets are to be replaced with numbers. This method permits a quick way of allocating fields of different sizes when program changes are needed. In the subroutine below forward and backward printing are performed using a Diablo 1650 or a Qume Sprint 5 with serial interface.

```
printing if reverse=0 then forwards else backwards
forwards print $0,buffers: reverse$: reverse=1: return
backwards for i=bufiz to 1 step -1: print $0,bufbyte(i): next i
print $0: reverse=0: return
```

Please be aware that bufbyte(0) is not printed on the backwards pass because the bell would ring. Therefore bufbyte(0) should always be a space. String assignment to fields must be done with LSET or RSET depending upon desired justification. The program has been tested on the SWTPC 6800 and the SWTPC 6809.

Sincerely yours,

*Manfred Peschke*

Manfred Peschke

681 Whitcomb Rd., #207  
Detroit, MI 48263  
March 18, 1980

Don Williams  
68 Micro Journal  
1010 Hamill Road  
Hixson, TN

Dear Don,

Here's a couple of little commands written for TSC's FLEX 2.0 DOS that some of your readers might be interested in.

If you're like me, you make a lot of use of FLEX's TTYSET command to switch the output environment when routine output alternately to the CRT and a printer. My personal preference is for a CRT screen depth of 20 lines (I have a 24 x 80 terminal), after which I like a pause so I can read it. This means for CRT output, I need TTYSET,DP=20,PS=V. Because of this, I must change the depth and pause parameters before routing output to the printer, where I want TTYSET,DP=0,PS=M. It got to be very tiresome after a while, having to input all those TTYSETs. I finally realized that knowing the locations of the TTYSET parameters in memory, it should be a simple matter to write a machine-language program that could be invoked as a .CMD, that with one or two characters of input from me (the command name), would set these memory locations to the desired values.

The result is LPR.CMD and RET.CMD for "Go to Line Printer mode" and "Return to CRT mode", respectively. Now I simply prefix and suffix any printing command with these two commands. (Just)

LPR:P,DATE:P,CAT:1:RET

Which (incredibly) used to be:

TTYSET,DP=0,PS=M:P,DATE:P,CAT:1:TTYSET,DP=20,PS=V

The addresses defining these various parameters can be found in the Flex User's Manual, in the "Advanced Programmer's Guide" or "Programmer's Manual" section (e.g., DOS MEMORY MAP).

AS - @AC00	OL - @AC01	EL - @AC02	AP - @AC03	LD - @AC04
NL - @AC05	TS - @AC06	OE - @AC07	EJ - @AC08	PS - @AC09
ES - @AC0A				

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

March 5, 1980

Use this command before any printer-related command.

It will set up the TTYSET parameters as required by the Heathkit H-14.

When operated at 300 BAUD, that is, TTYSET,NL=0,DP=0,PS=M.

For editing.

TAB 8: 13 30

ARM LPR

EQUATES

ARM5 EQU @AC05

ARM6 EQU @AC06

ARM7 EQU @AC07

ARM8 EQU @AC08

ARM9 EQU @AC09

ARM10 EQU @AC0A

ARM11 EQU @AC0B

ARM12 EQU @AC0C

ARM13 EQU @AC0D

ARM14 EQU @AC0E

ARM15 EQU @AC0F

ARM16 EQU @AC10

ARM17 EQU @AC11

ARM18 EQU @AC12

ARM19 EQU @AC13

ARM20 EQU @AC14

ARM21 EQU @AC15

ARM22 EQU @AC16

ARM23 EQU @AC17

ARM24 EQU @AC18

ARM25 EQU @AC19

ARM26 EQU @AC1A

ARM27 EQU @AC1B

ARM28 EQU @AC1C

ARM29 EQU @AC1D

ARM30 EQU @AC1E

ARM31 EQU @AC1F

ARM32 EQU @AC20

ARM33 EQU @AC21

ARM34 EQU @AC22

ARM35 EQU @AC23

ARM36 EQU @AC24

ARM37 EQU @AC25

ARM38 EQU @AC26

ARM39 EQU @AC27

ARM40 EQU @AC28

ARM41 EQU @AC29

ARM42 EQU @AC2A

ARM43 EQU @AC2B

ARM44 EQU @AC2C

ARM45 EQU @AC2D

ARM46 EQU @AC2E

ARM47 EQU @AC2F

ARM48 EQU @AC30

ARM49 EQU @AC31

ARM50 EQU @AC32

ARM51 EQU @AC33

ARM52 EQU @AC34

ARM53 EQU @AC35

ARM54 EQU @AC36

ARM55 EQU @AC37

ARM56 EQU @AC38

ARM57 EQU @AC39

ARM58 EQU @AC3A

ARM59 EQU @AC3B

ARM60 EQU @AC3C

ARM61 EQU @AC3D

ARM62 EQU @AC3E

ARM63 EQU @AC3F

ARM64 EQU @AC40

ARM65 EQU @AC41

ARM66 EQU @AC42

ARM67 EQU @AC43

ARM68 EQU @AC44

ARM69 EQU @AC45

ARM70 EQU @AC46

ARM71 EQU @AC47

ARM72 EQU @AC48

ARM73 EQU @AC49

ARM74 EQU @AC4A

ARM75 EQU @AC4B

ARM76 EQU @AC4C

ARM77 EQU @AC4D

ARM78 EQU @AC4E

ARM79 EQU @AC4F

ARM80 EQU @AC50

ARM81 EQU @AC51

ARM82 EQU @AC52

ARM83 EQU @AC53

ARM84 EQU @AC54

ARM85 EQU @AC55

ARM86 EQU @AC56

ARM87 EQU @AC57

ARM88 EQU @AC58

ARM89 EQU @AC59

ARM90 EQU @AC5A

ARM91 EQU @AC5B

ARM92 EQU @AC5C

ARM93 EQU @AC5D

ARM94 EQU @AC5E

ARM95 EQU @AC5F

ARM96 EQU @AC60

ARM97 EQU @AC61

ARM98 EQU @AC62

ARM99 EQU @AC63

ARM100 EQU @AC64

END LPR This draft 3-5-80

(NO ERROR(S) DETECTED)

## SYMBOL TABLE:

BEGIN ARM5 LPR ARM6 IN ARM7 ARM8 ARM9 ARM10



'68' Micro Journal

79	ACD3 80 BF	SEND	BSR	POUT	Else, send char. to printer
80	ACD5 28 F6		BRA	LOOP	Loop until PDB
81		*			
82		*			
83		*			
84	ABD0		ORG	PRCD8	Aligned space; PRCD8-PRCE3
86	ACD8 37	PDBK	PSH B		Save BR
87	ABD9 F6 89 BC		LDA B	AC1A	Get AC1A Status register
88	ACDC 56		ROR B		Get b.2 (OCD) Data Carr. Detect
89	ACDD 56		ROR B		into carry bit.
90	ACDE 56		ROR B		0-present; 1-not present
91	ACDF 33		PUL B		Restore BR
92	ACE0 39		RTS		
94			END		

NO ERROR(S) DETECTED

SYNOPSIS TABLE:

ACIA	BOOC	LOOP	ACCD	PCHK	ACDB	PINIT	ACCB	POUT	ACE4
POUT2	ACES	SEND	ACD3	SEQ	ACF1				

Yours,

Ernest M. Schuch

Keith Alexander

## COMPUTERWARE SOFTWARE SERVICES

### SMALL BUSINESS ACCOUNTS PAYABLE SYSTEM

**Want to Know Where Your Money Went - and Where It Is Going?**

The key to a successful business is good management of cash flow! Computerware's Accounts Payable System helps a small business manage and track its cash liabilities by collecting vendor invoice data and reporting the business' cash commitments and payment history.

And the system is easy to use. General information is stored for each vendor. As an invoice from a vendor is received, the pertinent data is entered into the system. The vendor summary fields are automatically updated, keeping all totals current at all times. Payments are entered by invoice number, invoice date, or applied against a vendor's oldest invoice. Again, vendor summary fields are updated automatically. Now, with a simple report request or on-line inquiry, you can have a concise list of all your outstanding bills - or a summary of a vendor's account - or a report showing how much you've spent this year, and how much you've paid - or a list of all your current payments. . . . or . . . In other words, you can see how you've been spending your money, how well you've kept your credit commitments, and what you will need to continue to meet your business' needs. You're in CONTROL!!

Information maintained for each vendor include name, address, phone number, terms extended, current balance due, total of invoices received during the current period, total amount paid during the current period, year-to-date paid total, and last activity date. The system stores the invoice number, vendor, invoice date, invoice amount, invoice type, invoice remarks, date paid, amount, and payment document number for each invoice entered. Reports include Account Cross Reference, Account Master, Account Summary, Accounts Past Due, Payment Forecast, and Activity. They may be sorted by vendor number, vendor name, invoice date with range selection available for selective reporting.

This system was designed to run on a 40K 6800/6809 computer with a minimum of a dual 5 1/4" disk system. Both the S88 D06 and Computervare's Random BASIC are required. Versions using other disk operating systems are pending customer demand. The Accounts Payable System is available immediately for \$149.

A manual describing the functions and operation of the system is available for \$15.00 from COMPUTERWARE - 1512 Encinitas Blvd. - Box 668 - Encinitas, Calif. 92024 - (714) 436-3512 or 436-0282.

**NAM**      **DISKSPEED**

```

* THIS PROGRAM WILL CHANGE THE HIGH SEEN SPEED
* FOR FLEX 2.0 DRIVING AS SUFFICIENT FOR THE M-AB
*
* IT WORKS BY CHANGING THE SEEN RATE AS SET ON THE
* 1773 FLOPPY DISK CONTROLLER IC.
* THE ORIGINAL RATE IS 40 MS PER TRACK.
* I TRIED USING 20 MS WILLI HX M-AB USING SIGNATURE SA-400'S
* AND HAVE NO PROBLEM.
* I DON'T KNOW WHAT WOULD HAPPEN WITH THE ORIGINAL WACOS'S
* AS THEY ARE SUPPOSED TO HAVE A SLOWER SEEN RATE.
*
* THE CHANGE IS IN A SUBROUTINE THAT IS THE VERY FIRST
* ROUTINE CALLED BY BOTH THE READ A SECTOR AND WRITE A
* SECTOR ROUTINES.
* THIS SUBROUTINE IS USED TO SETUP THE PROPER TRACK AND
* SECTOR AS CONTAINED IN THE A AND M REGISTERS.
* OF COURSE THE LOCATION WHERE THE TRACK SHOULD BE
* THE COMMAND LDAI $1B (06 1B).
* THE TWO LAST SIGNIFICANT BYTES OF THE $1B THAT THE
* DISK CONTROLLER WHAT TRACK SIZE: SPEED TO USE.
* A 0000 0011 IS FOR 40 MS WHILE CHANGING TO 0000 0010 IS
* THE MUCH FASTER 20 MS RATE.

```

FAGT	EDU	SLA
SLOW	EDU	DSH
BYTE	EDU	GREFF

Bill Vadell  
Box 336  
Ollmont, MT 59466

ANALOG  
DIGITAL  
MIXED

Thomas Instrumentation  
6800/6809 COMPUTERS



168-8th Street, Avalon, N.J. 08202  
(609) 967-4280

March 21, 1980

Dear Dan and Larry,

Until now, Thomas Instrumentation's main business has been industrial controllers, but 68 Micro Journal has changed us. We have never tried to re-invent the wheel...only when we couldn't buy cards off the shelf did we design our own. Within the last year we have seen many changes in the 68-50 market place, mainly due to our advertising in the 68 Micro, the 68-50 portion of our business has grown to the point of a viable market. For this reason I have promoted Donna to handle 68-50 sales and we are going to design many new products that have been requested. In addition, we will also stock RCA keyboards and Video 100-50 Monitors as well as SWTPC, GIMIX, TSC, and other compatible products.

Our own products now include SUPER CPU, VIDEO RAK, 10 port I/O, Wire-wrap, 3,4,6,12,16 slot backplanes, and Power supplies. We offer a small starter system (fully assembled), including RCA keyboards, Video 100 monitor, SUPER CPU, VIDEO RAM, backplane and power supply, for less than \$600.00. It is fully operational and can even be used as a terminal, as well as a development system.

It is our policy to sell fully assembled and tested cards, or bare cards for those who wish to "roll their own". We will also make available certain key parts that may be difficult to obtain. We will not offer kits of our cards, although we do stock SWTPC kits.

We are currently working on adding graphics to our video card, several chassis (including rack mount), more backplanes (30 pin and adapters), I/O cards (including A/D, modems, etc.), and 16k static ram card (using 2114). In addition, we have signed a licensing agreement with JFC Products Inc. to make a small add on card to our CPU that will duplicate their fine cassette system (CFM3) will be available in ROM on the CPU.

Thanks for a good year,  
Thomas Gluyas

TWG:df

MARCH 6, 1980

6800 MICRO JOURNAL  
3018 HAMIL LANE  
P.O. BOX 849  
HILSON, TEXAS 75743

GENTLEMEN:

THE ATTACHED ASSEMBLED LISTING IS SUBMITTED FOR PUBLICATION WITH THE HOPE THAT IT MAY BE OF SOME BENEFIT TO OTHER JFC (CFM3) 685 CASSETTE SUPER BASIC USERS. I UNDERSTAND THAT JFC AND CCS HAVE A NEW 685 SUPER BASIC-3 "IN THE WORKS" AND I CAN HIGHLY RECOMMEND THE PRODUCTS OF BOTH COMPANIES. IN THE MEANTIME, FOR CFM3 USERS, THE ATTACHED "TSAVE" AND "TLOAD" COMMANDS WILL REPLACE THE OLD 300 BAUD COMMANDS BY THE SAME NAME AND PERMIT LOADING OF SOURCE (PROGRAM) FILES AT 4800 BAUD.

SINCERELY,

Ernest Stein

RDWAV

SPECIFICATIONS FOR THE FOLLOWING ASSEMBLED LISTINGS WERE JFC'S CFM3 VLSI 2.0A CARD AT 4800 WITH STACK AT 5000-5.0FF AND CCS' SUPER CASSETTE BASIC VER. 4.3

PAGE 001 TSAVE

00010	00020	00030	00040	00050	00060	00070	00080	00090	00100	00110	00120	00130	00140	00150	NAME	TSAVE	
00020	0010														ORG	LOC10	
00030	A300	CPSAVE	EQU													LA200	(ANY COMMENT TO BE PLACED)
00040	0020	BEGSC	EQU													L20	PRINTED FOR REG. IF PROGRAM
00050	0022	ENDSC	EQU													\$22	PRINTED FOR END OF PROGRAM
00060	0002	EXSAVE	EQU													1C002	CFM3 ENTRY FOR ASCII TO DEC
00070	0010	FF	A200	STX												CPSAVE	SAVE INDEX REGISTER
00080	0020	DI	A200	LDR												BEGSC	
00090	0022	FF	A002	STX												LA002	BEG. PUNCH POINTER
00100	0015	DE	A2	LDR												ENDSC	
00110	0027	FF	A004	STX												LA004	END. PUNCH POINTER
00120	002A	FE	A200	LDR												CPSAVE	RESTORE INDEX REGISTER
00130	0030	WH	C002	JSR												EXSAVE	AND JUMP TO CFM3
00140	0030	39		RTS													
00150				END													

TOTAL ERRORS 00000

NO ERROR(S) DETECTED

```

1  * USE THE FOLLOWING BLOCK OF JFC'S AND COME
2  * TO LINK WITH THE FLEX.COR SUPPLIED BY
3  * JFC IN ORDER TO EXPRESS THE DATE PROMPT
4  * RECEIVED WHEN STARTING FLEX.
5
6  * THIS IS A MODIFIED VERSION OF THE ROUTINE GIVEN
7  * ON PAGE 27 OF THE ILL GENERAL NOTES IN THE USER'S
8  * MANUAL.
9
10 * THIS WORKS WITH THE FLEX.COR I RECEIVED
11 * FOR A SWTPC 6800 MINIFLOPPY DISK SYSTEM.
12 * HOWEVER, I ADDED THAT THE ADDRESSSED
13 * WERE DIFFERENT ON THE FLEX.SYS
14 * THAT WAS INCLUDED WITH THE SAME SYSTEM.
15 * THE GET DATE ROUTINE WAS AT $A6A7 RATHER THAN $AA67,
16 * DUE TO THE $A7C INSTEAD OF $AA67 AND THE DATE STRING
17 * WAS AT $A0D2.
18
19
20 * AFTER ASSEMBLING THIS BLOCK, DO THE FOLLOWING:
21
22 * $APPEND FLEX.COR,AUTODISK.DIR,FLEX.SYS
23 * $FILE FLEX.SYS
24
25 * THE SYSTEM SHOULD NOW BOOT WITHOUT THE DATE PROMPT.
26
27 * USE THE DATE UTILITY ROUTINE IF IT IS
28 * NECESSARY TO CHANGE THE DATE.
29
30 $A000 XFR EQU $A000 D05 XFR ADDRESS
31
32 BEA3 ORG $BEA3
33
34 BEA3 E1 AC INVEC FDB $E1AC
35 BEA5 E1 D1 DIVEC FDB $E1D1
36 BEA7 00 04 ACIA FDB $B004
37 BEA9 00 12 TIMEK FDB $B012
38 BEAB 00 00 IRG FDB $A000
39 BEAD 00 12 SWI FDB $A012
40 BEAF 00 00 ROM FDB $E000
41 BEB1 00 4B PCV FDB $A04B
42
43 * MODIFICATION TO SKIP DATE
44
45 $AA67 ORG $AA67
46
47 $AA67 DE AE 0B LDX $STRING DATE STRING
48 $AA69 BE AE 9E JSR $BEVE PRINT STRING
49 $AA6B 06 07 LBA A 07 JUMPY DATE
50 $AA6D 07 AE 0E STA A $AC0E MONTH
51 $AA6F 07 AE 0F STA A $AC0F DAY
52 $AA75 07 AE 10 STA A $AC10 YEAR
53 $AA7B 7E AA 05 JMP DONE
54
55 $AAB5 DONE EQU $AAB5
56
57
58 * THE DATE STRING MAY BE MODIFIED TO PRINT WHATEVER
59 * SIGN-ON MESSAGE IS DESIRED.
60 * THERE IS ROOM FOR A 17 BYTE STRING
61 * THAT MUST BE FOLLOWED BY A $04.
62
63
64 $AAB9 ORG $AAB9
65
66 $AAB9 STRING EQU *
67 $AABD 42 FCC 'BY YOUR COMMAND..
68
69 $AAC0 59 20
70 $AAC2 59 4F
71 $AAC4 55 52
72 $AAC6 20 43
73 $AAC8 4F 40
74 $AACB 40 41
75 $AACD 4E 44
76 $AAE0 2E 2E
77
78 $AAC0 04 FCC $04
79
80 END XFR

```

NO ERROR(S) DETECTED

SYMBOL TABLE:

ACIA	BLA7	DONE	AAB5	INVEC	BEA3	IRG	BEA9	ROM	BEAF
OUTVEC	BEA5	PCV	BEB1	STRING	AA0B	SWI	BEAD	TIMEK	BEA9
XFR	AD0B								

```

00010      HAI TL8AD
00020 0C72  RAO 30C72
00030      C005 EXL8AD E0H 30C05
00040      0022 EHU5C ECU 32
00050      0022 STATUS ECU 3A087
00060      0782 OK ECU 3C782
00070 0C72 8D C05 JSH EXL8AD
00080 0C75 F2 AF8F LUX STATUS
00090 0C78 6C C782 CPX 38K
00100 0C7B 26 0F HNE DUMP
00110 0C7D FE A104 LDX 3A004 ENO. PUNCH POINTER
00120 0C80 D7 25 STX ENDSC
00130 0C82 39 RTS
00140 0C83 37 DUMP 34004 ABORT LOAD
00150      END EUD

```

TOTAL LENGTH 00000

STEVE MAHER  
2949 CEDAR AVE. S.  
MINNEAPOLIS, MN 55404  
(612)-722-1702

FEB. 27, 1980

Mr. Don Williams, Sp.  
1014 HAMIL ROAD  
NIXSON, TENN. 37343

DEAR FID:

I HAVE A PROGRAM WHICH MAY BE OF INTEREST TO PEOPLE WHO ARE CURIOUSLY. IT WILL RUN ON A BUILDING UP A BASIC SYSTEM.

IT IS A 2-PART OF THE BASIC-ASSEMBLER, WRITTEN IN STANDARD BASIC. IT CAN BE RUN ON ANY LARGE OR SMALL COMPUTER, USING INTERPRETER. THIS CAN BE VERY HANDY FOR THOSE WHO DO NOT YET HAVE THE BASIC UP AND RUNNING. IT PROVIDES WORKING SOLELY WITH INTERPRETER, ETC. IT SUPPORTS ALL THE BASIC INSTRUCTIONS AND ADDRESSING MODES, AND WILL ALSO TRANSLATE ALL THE INSTRUCTIONS FROM THE BASIC, FOR THE BASIC, ETC. INSTRUCTION SET, INTO BASIC CODE AND ASSEMBLY.

ANYONE INTERESTED IN THIS PROGRAM SHOULD SEND A SELF-ADDRESSED, STAMPED ENVELOPE TO THE ABOVE ADDRESS. LET THE JOY BE UNIVERSAL.

SINCERELY,

*Steve Maher*  
STEVE MAHER

P. S. I'M ALSO WORKING ON A SIMILAR PROGRAM FOR THE 16-BIT M6800 MICROCOMPUTER. STAY TUNED FOR FURTHER DEVELOPMENTS....

March 6, 1980  
2521 W. Hill Road  
Delatland, WI. 53018

'68' Micro Journal  
P. O. Box 849  
3018 Hamill Road  
Nixon, Tenn. 37343

Dear Persons:

It has not been my practice to write letters to publications, but I feel that I have reason now to give it a try.

As an Amateur Radio operator and electronics experimenter for many years I was extremely surprised at how inadequate and lost I felt when initially exploring the area of micro computers. Starting cold, it took me about six months of reading before I felt comfortable enough with the jargon to begin talking seriously with professionals about their products. Frankly, some of the responses I got might have turned me away if the bug had not bitten so bad that I persisted. Then, thanks to Dave Weeks of F & D Associates, I was introduced to the 9850 Buss. My experiences with "professionals" began to change immediately, in a positive direction. I was accepted into a group of persons who seemed equally interested in pleasure of experimental learning and developing functional micro systems, and sharing both with others. Now that I have been reading '68' Micro Journal for 7 months that feeling of fraternity has become even greater. I want to thank '68' Micro and everyone else for helping me into a valuable learning experience and a pleasant, challenging hobby.

Since I have not seen F & D Associates mentioned before in '68' Micro, and since my experiences with that company have been very satisfactory, I thought I might tell you a little about it. I will also enclose a catalog of products in case you have never received one. The products (from CPU Boards to video boards to I/O Cards) are of excellent quality. The documentation has been very adequate, even for me. But most important has been their willingness to help the user. Dave Weeks and I have written so many letters, we are practically pen pals. I am very sure that many others would benefit if they write to Dave at F & D, 1210 Todd Road, New Plymouth, Ohio, 43654.

Thank you for this opportunity for expression and for a super Journal.

Sincerely,  
*Jim Gettred*  
Jim Gettred  
WAS925G

P.S. I would sure like to hear from folks who know of sources of Amateur Radio RTTY, CW, etc. software for 6800 users. I don't think the other busses have a monopoly on that -- I hope.

Country Club Road  
Horseshoe Junction, NY 12533  
March 10, 1980

'68' Micro Journal  
3018 Hamill Road  
Nixon, Tennessee 37343

Dear Mr. Williams,

I would like to offer a few comments relative to several items from the March '80 issue of '68' Micro Journal.

COMPUTERFARE ... re: Paul Searby's letter

I encourage Computerfare to offer their software in FLEX as well as SIB. My MINIFLEX version of CCS has been purchased about a year ago, works very well, and has a few more features than SMTTC disk BASIC. (It seems, in reviewing the latest Computerfare flyer, that this is no longer offered, which is too bad.) If Computerfare's business software is comparably good, it should be well received by FLEX owners. FLEX DOS is hard to beat, which leads me to ...

TSC ... re: letters from John Tucker and Don Canada

I have nothing but praise for TSC for quality in both software and documentation. The several queries I have made have resulted in prompt and helpful responses. The TSC 6800 BASIC is fast and versatile. I have not bought the Extended BASIC yet, but probably will. However, I am left with the uneasy feeling that there are known bugs (but not by me!) in my TSC BASICS. I feel sure that if I run into problems I will get help from TSC, but I would feel a lot better if I could buy an "insurance policy" which would ensure prompt notification of bugs and fixes. For major changes, such as those which would require a disk reformat, an additional charge would be justified. The key point is that the user should not be left to find each and every problem on his own.

Finally, in order to contribute in some more tangible way to the 6800 world, I offer the following short BASIC routine which provides for a quick assessment of the randomness of RNR. This program runs as written with my SMTTC, CCS and TSC BASIC interpreters. The request for "RANDOMITY" is as in: for the number of discrete integers which comprise the set from which one is selected on each trial.

```

0010 INPUT "RANDOMITY",G
0020 DIM X(G)
0030 INPUT "NUMBER OF TRIALS",N
0040 FOR J=1 TO N
0050 LET Y=INT(G*RND(0)+1)
0060 LET X(Y)=X(Y)+1
0070 NEXT J
0080 PRINT
0090 PRINT "AFTER ";N;" TRIALS:"
0100 PRINT
0110 FOR I=1 TO G
0120 PRINT I;"SELECTED ";X(I);" TIMES"
0130 NEXT I
0140 END

```

Best wishes and my thanks to you, the staff, and the many contributors to '68' Micro Journal. It continues to give the best price/performance in the field of microcomputing literature.

Yours very truly,

*E. D. Council*  
Edwin D. Council III

\* FLEX is a trademark of TSC, Inc.

33 McKinley Avenue  
West Caldwell, N.J.  
07006  
March 6, 1980

Mr. Don Williams  
'68' Micro Journal  
3018 Hamill Road  
PO Box 849  
Nixon, Tennessee 37343

Dear Mr. Williams:

Many thanks for the diskette of Stock's Bookkeeping program. I have not tried to run it yet since MENU.BAS appeared to be the same as START.BAS. I didn't write you before since I figured that I could copy the real MENU.BAS from the magazine when it came. However, it came and seems to have the same trouble.

I hope I am not the 539th person to write you about this problem. Please don't let this deter you from continuing to publish applications type software.

Sincerely,

*George Field*  
George Field



the lines are numbered by 10's. After the listing for the modification of Print has been entered, the jump address in the table must be changed from 0247 to 0497. Also, starting at address 0297 change the next three bytes of code to 7E 041E. If you are using Mikbug you need to change the code at address 0287 from PA to P8. This change will allow all spaces to be outputted.

The other reason for changing Chuck's program was because of another machine language Editor written by Peter Stark in the January, 1979, issue of Kilobaud on page 22. In his editor there is a Find command. It will find all occurrences of a certain string. This Find command sounded interesting so I thought I would try to adapt Pete's program into Chuck's. The nature of Chuck's string storage made the adaptation difficult but not impossible. Also, the abbreviated form of the Insert command in Chuck's editor made adaptation difficult. The format of the Find command is: `F/string/`. The slashes are delimiters that can be any character or punctuation not used in the string itself. The string can include imbedded spaces but not spaces on the end of the string. The imbedded spaces caused a bunch of problems when I tried to adapt the Insert function for use in the Find command. The string can be as short as one letter or punctuation character, not including the space. It can be as long as 80 characters. All occurrences of the string will be printed. I didn't try the Find command on anything very long but for what I did use it on Find seemed instantaneous. After the listing for Find is entered you must put it in the jump table somewhere. Since my cassette interface was in the shop I replaced the Load command. At address 013C I changed the 0400 to 0497. Other patches are noted in the listing.

One last note, Peter Stark's program has a Replace command if you want to make this into a character editor. I hope you will excuse any errors and amateur style. This is the first time I've used my TSC Assembler and Heath printer for something serious.

```

* LINE EDITOR MODIFICATIONS
*
* MODIFICATIONS INCLUDE:
* PRINT A SET OF LINES (P L8,L8)
*
*
*
* OPT PAG.NHM
* ORG 40497
*
0078 FID EQU 40028
0079 OUTSP EQU 40071
007A INHR EQU 40029
007B TEMPOR EQU 40030
007C DEL1 EQU 4003C
007E CAL2 EQU 4003E
007F STRMEM EQU 40032
0311 DELF EQU 40011
0026 BEGIN EQU 40026
0267 PRINT EQU 40267
0076 SHUX EQU 40076
02A2 PRINTS EQU 402A2
*
*
0497 CE 00 00 LDX #0
0498 DF 28 STX END
049C 80 00 71 JSR OUTSP
049F 80 03 29 JSR INHR
04A2 81 2C CMP A #7
04A4 26 33 RNE GORINT
04A6 DF 3A LDX TEMPOR
04A8 DF 3C STX DEL1
04AA 4F CLR A
04AB 80 03 29 JSR INHR
04AF 4F 38 LDX TEMPOR
04B0 DF 3E STX DEL2
04B2 96 3F LDA A DEL2+1
04B4 C6 0A LDA B #00A
04B6 4B RRA
04B7 24 05 SCC NOC
04B9 7C 00 3E INC DEL2
04BC 97 3F NOC STA A DEL2+1
04BE DE 32 LDX STRMEM
04C0 96 3C LDA A DEL1
04C2 D6 30 LDA B DEL1+1
04C4 80 03 11 JSR DELE6
04C7 DF 26 STX BEGIN
04C9 4E 32 LDX STRMEM
04CB 96 3E LDA A DEL2
04CD D6 3F LDA B DEL2+1
04CF 80 03 11 JSR DELE6
04D2 DF 28 STX END
04D4 DE 26 LDX BEGIN
04D6 7E 02 67 JMP PRINT
04D9 DE 32 GORINT LDX STRMEM

```

## 68'FORTH

- COMPILER
- INTERPRETER
- EDITOR
- DICTIONARY DRIVEN
- EXTENSIBLE VOCABULARY
- VIRTUAL MEMORY

FAST — KILOBAUD (Oct '77) benchmarks #1 and #7 in 0.07 and 8.5 seconds (mixed 16 and 32 bit math).

FORTH™ is an interactive language ideal for data collection and analysis, instrument control, graphics, and algorithm development. Gives assembly-level control of machine with only 10-30% of development time — yet is also a high-level compiler. Compiled code is 2½-14 times faster than TSC BASIC.

Full implementation of 1978 FORTH Interest Group standard for 6809 with dictionary names to 31 characters, 16 and 32 bit math, compiler error checking, and source text editor. Minimum memory requirement: 12k (20k if wish to interface with FLEX™). 16k recommended. Manual contains tutorial for language, description of supplied vocabulary words, and information for interfacing with any disk system. Supplied for SWTP 6809 and MF-68 5" dual disk with FLEX™ 9.0; can be modified for other disk systems using cassette medium to make initial load, and in-ram simulation of disk. Tape-only is possible too.

Visa, Master Charge, cashier's check, or money order — Texas residents add 5% tax — shipping and handling: US-included, foreign—add \$3.00.

Talbot Microsystems  
7209 Stella Link Suite 112 M Houston, TX 77025  
Phone (713) 667-9855 ext.112

FORTH is trademark of FORTH, Inc.; FLEX is trademark of Technical Systems Consultants.

300 BAUD KCS cassette	\$89.95
5" disk for MF-68	\$39.95
Documentation alone	\$3.25
(deductible when system ordered later)	

### FORTH for 6809

**A-VIDD**  
electronics co.

2210 Bellflower  
Boulevard  
Long Beach, CA  
90815

Hours:  
Mon - Thurs 8:30 AM-5:30 PM  
Fri 8:30 AM-9:00 PM  
Saturday 10:00 AM-5:30 PM



SPL/M is a block-structured language which features arbitrary length identifiers and structured programming constructs. It is suitable for systems programming on small computers, since the compiler requires only 20K of memory and a disk system. SPL/M is a pure code compiler and is currently available for 6800 computer systems using either FLEX II or SSB's DOS68 51 disk operating systems. Package consists of 3 SPL/M Library files which allow both terminal and the I/O. Most Major DOS routines are supported.  
Price for FLEX II or SSB format \$49.95

### SPL/M for FLEX II

The SD Compiler Basic is the most well developed basic for the 6800. Some of the more notable features include: Formatted Print Statements, If Then Else & While Do, Variable Names Up To 15 Characters and High Speed Execution. Both random and sequential device I/O can be done in either binary or ASCII mode for data flow control to the byte. Now available for FLEX II, FLEX I, MINIFLEX and SSB FLEX II. Package includes: Basic compiler, Mail assembler (with extensive manuals for each), run time package and 4 misc. utilities. Call or write for detailed catalog. Dealer inquiries invited.  
Price \$330.00  
Software Dynamics Editor \$100.00

### 6800 PRODUCTS AT A-VIDD

Software Dynamics Compiler Basic



# Inventory Problems?

Are you having trouble keeping the right nuts and bolts in stock? Since even a simple mistake can cost you time and money, a good inventory system should do more than just count parts. It should tell you exactly what you need, when you need it, where to get it, and how much it will cost.

The MSI Inventory System Seven enables you to maintain a versatile data base for controlling inventory. It lists part number, description, quantity on hand, vendor, cost, selling price, optional pricing, usage levels for previous month, present month, and year-to-date, and much more.

When quantity on hand items reach minimum levels, the System Seven compiles an automatic reorder list. This list can be generated by specific vendor as well as a complete listing of all materials to be ordered.

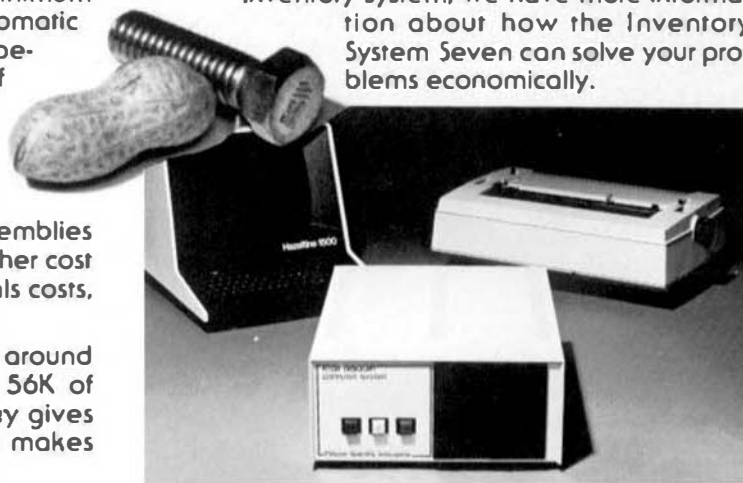
In addition to the item listing, the Inventory System Seven "bill of materials" provides you with a complete inventory of items used in the manufacture of subassemblies and complete products. It also contains other cost items such as labor costs, total raw materials costs, and miscellaneous costs.

The MSI Inventory System Seven is built around the versatile MSI 6800A Computer with 56K of RAM. An integral dual mini-floppy memory gives you an additional 630K of memory and makes

Inventory control fast and efficient. The System Seven will interface with any industry standard CRT, and you have the option of both a "daisy wheel" word processor for high quality document preparation and a dot matrix printer for high speed production.

The System Seven can be expanded to handle all your data processing needs or you can select one of nine other MSI systems now available for business, industrial, scientific, educational, and personal applications.

If you need more than just a nuts and bolts inventory system, we have more information about how the Inventory System Seven can solve your problems economically.



**MSI Inventory System Seven**

# MSI

# Midwest Scientific

220 W. Cedar, Olathe, Kansas 66061, (913) 764-3273  
TWX 910 749 6403 (MSI OLAT), TELEX 42525 (MSI A OLAT)

## 6800 Software Hardware, Firmware

This month's Special...

### BASIC - UP

The Basic Utility Package is a set of six programs to make programming in Basic a snap. With Basic-UP you can renumber your programs, generate neat and readable printouts by "pretty-printing", get an index of all variables or jumps and where they occur, compare two programs and print out only the differences between them, shorten program size and speed up execution, and more. Truly a Basic Programmer's Dream. Available on miniFlex (TM of TSC) disk for SWTP Basic or on Percom LFD-400 Disk for \$30 list -- this month's special price is \$25, which includes a free copy of FLDGEN. Specify which Basic and DOS you use. Special - get a copy of Eliza for just \$5 more.

### STAR - KITS

P.O. Box 209, Mt. Kisco, N.Y. 10549

### 6809 — DATA FILE MAINTENANCE

STOP writing dinky little programs for all the one-time changes to one item on a data file. START bringing up new systems without long weeks of programming.

The General Data File Maintenance Program can add, delete, insert, and modify data on any file you have\*! The powerful security allows you to restrict modification of data already entered.

This software tool will save you days of programming effort with commands that can list, print or show your data. Some of the many things you might use it for are: Inventory files, Customer files, Real Estate Listings, plus many more. Let your imagination run WILD!

You can format the items in many ways with this 6809 Extended BASIC program. Some of the options available are right or left justify, item length, etc.

Order your diskette today for only \$49.95! Use Master Charge, VISA or check. Specify diskette size.

Tennessee residents add 6 1/4% sales tax. Customers outside Canada or USA add \$5.00 for air postage and handling. If you wish to order by phone, give us a call at (615) 396-9111.

Coming soon VER. 2.1 for TSC Multiuser Basic™

\*Record sizes up to 252 bytes.

# dp systems

po box 567

collegedale tn, 37315

### MULTI-USER ON A 56K 6809

Support four totally independent terminals with one computer. Each terminal has access to the services you need most in your classroom or business.

**COMMON PILOT** - the best CAI system in the world; useful in any application where computer-user dialogue is desirable.

**STUDENT BASIC** - real numbers, functions, arrays, strings, loops, subroutines, and random disk-file access.

**SCREEN-ORIENTED EDITOR** - edit lines directly by overtyping, inserting, and deleting; useful for programming and word processing.

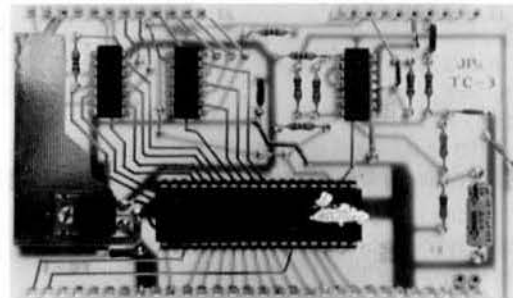
Hardware requirements: SWTPC /09 (the cheaper 6809), 56K, MP-N calculator board, dual floppy disks (either size), one to four MP-S serial interfaces, one to four SWTPC CT-82 terminals.

\$250.00 single copy (specify 5" or 8")  
(quantity discounts available)

**MICROPI**  
2445 N. Nugent  
Lummi Island, Washington 98262

### JPC PRODUCTS FOR

# 6800 COMPUTERS



### High Performance Cassette Interface

- **FAST** - 4800 Baud Loads 4K in 8 Seconds!
- **RELIABLE** - Error Rate Less Than 1 in 10<sup>6</sup> Bytes.
- **CONVENIENT** - Plugs Directly Into The SWTPC.
- **PLUS** - A Fully Buffered 8 Bit Output Port Provided.
- **LOW COST** - \$59.95 For Complete Kit.

- **OPTIONAL** - CFM/3 File Manager.  
Manual & Listing \$19.95  
(For Cassette Add) \$ 6.95

TERMS: CASH, MC or VISA. Shipping & Handling \$1.00



Order Phone (505) 294-4623  
P.O. Box 5615  
Albuquerque, N.M. 87185

# COMPUTERWARE

## is serious about

### 6800 / 6809

#### System Software

DOS/Utilities  
MONITOR  
XREF (cross ref.)  
Assembler

Random BASIC  
BASREF (cross ref.)  
RENBAS (renumbering)  
Editor

#### Application Software

Accounts Receivable  
Accounts Payable  
Ledger Accounting  
Inventory Control  
Random Data Organizer / Report Generator

Payroll  
Medical Office  
Word Processing  
Mailing System

all available for both 6800 and 6809  
Applications for home, small business, and commercial users

#### And we have the hardware too!

Smoke Signal Broadcasting • SWTPC  
Centronics • NEC • Anadex • SOROC  
Micro Works • Thomas • Newtech • Sanyo

Call, Write, or Come See Us at:  
**COMPUTERWARE**  
1512 Encinitas Blvd. Box 668  
Encinitas, CA 92024  
(714) 436-3512

## P-6800 PASCAL



RELEASE 2

P-6800 PASCAL now has REALS (10<sup>-30</sup> digit precision)  
P-6800 PASCAL now has RECORDS (owntype definitions)  
P-6800 PASCAL now supports RANDOM FILES<sup>™</sup>  
P-6800 PASCAL now has a COMPLETELY REVISED MANUAL  
P-6800 PASCAL includes lots of SAMPLE DEMO PROGRAMS  
P-6800 PASCAL has a SIMPLE USER interface  
P-6800 PASCAL is easily linked to your own peripherals  
P-6800 PASCAL does NOT require you to buy more hardware  
P-6800 PASCAL will still run in ONLY 16K  
P-6800 PASCAL requires ONLY ONE SINGLE DENSITY 5" DRIVE  
P-6800 PASCAL requires NO SPECIAL TERMINAL  
P-6800 PASCAL does NOT force you to abandon FLEX<sup>™</sup>  
P-6800 PASCAL STILL costs only \$350 (VISA/Mastercharge)

P-6800 PASCAL is supplied in versions 2.1 minifLEX  
2.2 FLEX 2  
2.9 FLEX 9 (6809)

#### FURTHER DETAILS FROM:

LUCIDATA  
Opsteinde 223  
2271 EG VOORBURG  
NETHERLANDS

<sup>™</sup> FLEX is the trademark of TSC Inc.  
\* Not in Version 2.1 (minifLEX)

•••••

### 6800/6809 SOFTWARE

#### ACCOUNTS PAYABLE #1300

Produce financial reports, print checks, special control letter. Reports by vendor number, invoice number, aged and history file. Auto sorting of vendor and invoice files. Plus check and pre check journals. \$600.00

#### ACCOUNTS RECEIVABLE #1500

Produce financial reports, prints statements, produces reports by customer account number, invoice by customer account number and invoice by invoice number. Print aged report and final balance. Keeps history file and auto sorting of files. \$800.00

#### GENERAL LEDGER #100

Program updates to ledger files and also generates reports on payroll, sales, accounts payable, cash and expense statistics. Balancesheet and profit & loss reports. Information can be generated for year end taxes, 941 and W2 forms. \$595.00

#### INVENTORY II #700

Produce inventory reports by description or vendor, print activity reports for one day, one month or one year. Quick search by part number, produce total inventory and financial report. (For one store) \$200.00

#### MAILING LABELS #100

Print mailing labels from your complete file, for a particular city or state. Use one-part mailing labels. \$ 50.00

#### MAILING LABELS #400

Same as #100, but also prints labels by names. Use multiple-part labels.

#### BASIC—0935

For those with applications in SWTP 3.5 BASIC. Runs on 6809 and 6809S. 30% faster and can be used with existing 6800 BASIC programs. No manual commands and statements same as SWTPC BASIC 3.5 Ideal to keep you going while changing to new BASIC's 5¼ or 8 inch 09 Disk, with renumber routine. \$ 59.95

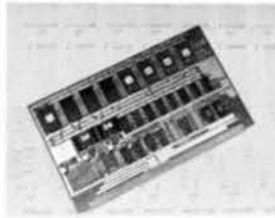
Available from Computer Stores or order direct from:  
Omni-Tronics Inc, 1897 Rt. 33, Concord Square,  
Hamilton SQ., NJ 08690  
Phone 609-890-9197

••Customized programs for your business requirements••  
Charge your order to your Visa or Master Charge

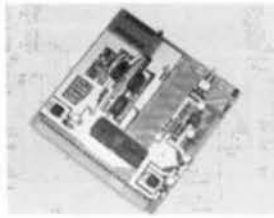




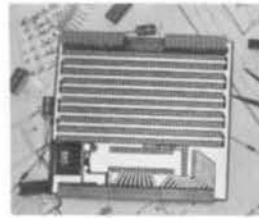
DS-68 DIGISECTOR



PSB-08 PROM SYSTEM BOARD



B-08 EPROM PROGRAMMER



UIO UNIVERSAL I/O BOARD

## INNOVATIVE PRODUCTS FOR 6800 USERS

**DS-68 DIGISECTOR** is a random access video digitizer featuring 256 x 256 picture element scan and 64 levels of grey scale, with conversion times as low as 3 microseconds per pixel. It accepts either interlaced (NTSC) or non-interlaced (industrial) video input. Use it for computer portraiture, moving target indicators, precision security systems, fast to slow scan conversion... with clever software, the Digisector can read just about anything. Truly a professional tool at a price you can afford. \$169.95

**B-08 2708 EPROM PROGRAMMER** is a compact unit that fits in the 6800's I/O slot. A safety switch and LED indicator provide control over the high programming voltage generated on board. An industrial quality Textool socket and extended board height allow effortless PROM insertion and retrieval. Fully commented source listings of U2708 is included in the Owner's Manual. \$99.95

**U2708** utility for testing, burning, verifying and copying 2708s in EPROM. \$29.95

**PSB-08 PROM SYSTEM BOARD** features 1K of high speed, low-power RAM and space for up to 8 2708 EPROMs, both DIP-switch addressable to start on any 8K boundary in memory. The exclusive I/O select feature allows you to move I/O locations up to any unused 1K block in the EPROM memory space. This permits memory expansion to a full 56K of contiguous user RAM. \$119.95

**DM-85 DISK MIXER** is an add-on board for the Smoke Signal Broadcasting BFD-68A Disk Controller which allows operation of both 8" and 5" drives. Controller mode (8" or 5") is selected on a drive-by-drive basis, so any mix of 5" and 8" drives is allowable. The 2" x 3" PC board mounts inconspicuously on the back of the BFD-68A. Its operation is completely transparent to software. An oscilloscope is required for the setup procedure. Kit Price: \$39.95

**M6809 EMULATOR** is a machine language program that will emulate all of the functions of the Motorola 6809 third generation microprocessor. Developed for use on any 6800 system, the program allows software development and debugging. The 3K byte program is complete with a 6809 mini-monitor and single-step trace routines. Fully commented source listing included. Specify Smoke Signal Broadcasting or FLEX™ disk, or KCS cassette. \$49.95

**UIO UNIVERSAL I/O BOARD** helps you with your custom interfaces. It has space for a 40-pin wire wrap socket into which you may plug any of Motorola's 40 or 24-pin interface chips. All data and control lines are connected to the appropriate edge connector pins. All other bus connections are brought out to a 16-pin socket pad. +5 volt regulator and all Molex connectors are provided; regulated +5 and ground are bused among the locations for up to 35 14-pin ICs. \$24.95

THE **MICRO  
WORKS**

P.O. BOX 1110, DEL MAR, CA 92014 714-942-2400

## TIRED OF PLAYING THE QUESTION AND ANSWER GAME?

Frustrated by the limitations of BASIC? Tired of question and answer data entry? Ever wish you could enter and update data the way the **BIG SYSTEMS** do it? Irritated by having to re-enter someone's entire name when all you wanted to do was change one letter? **WAIT NO MORE! NOW YOU CAN**

### CONFORM

**ALFORD AND ASSOCIATES** is proud to present the biggest little item to hit the small systems world since **BASIC** itself. **A & A** has produced a program that allows users of **MEMORY-MAPPED VDU'S** to display a data-entry form on-screen. Fill in the blanks, make **SCREEN-EDIT** changes with functions such as **INSERT SPACE, DELETE CHARACTER, ERASE TO END OF FIELD, ERASE FIELD, TAB FIELDS, CLEAR ALL DATA FIELDS**, for new entry, etc., and do it all under the control of your **BASIC** program!

### CONFORM works with

**ANY POPULAR BASIC - TSC, COMPUTERWARE, SWTPCO, ETC. . . .**  
**ANY POPULAR DISPLAY - GIMIX, PERCOM, SSB, THOMAS, ETC. . . .**

Why play the question and answer game with your Basic program? **CONFORM** with **A & A** **CON**trolled **FORM**s data entry Basic overlay program today! Available now on disc or cassette for only

**\$24.95**

including program media, source listing, sample programs and manual

**ALFORD AND ASSOCIATES, P. O. BOX 6743, RICHMOND, VA., 23230-804-329-3906**

Shipping and handling extra on orders under \$100.00/a. Residents add 4% Sales Tax. **UPS COD, VISA, MASTERCHARGE**, personal checks all accepted graciously.

## THE SCREDITOR SCREEN EDITING UTILITY IS NOW EVEN BETTER!

### CHOOSE ONE FROM COLUMN A AND ONE FROM COLUMN B

A (DOS)

B (VDU)

SSB DOS 4.X

GIMMIX 80x24

SSB DOS 5.X

GIMIX 64x16

MINIFLEX

SSB VDB-1

FLEX 1.0

THOMAS INSTRUMENTATION

FLEX 2.0

PERCOM

### CHOOSE ONE FROM COLUMN A AND ONE FROM COLUMN B

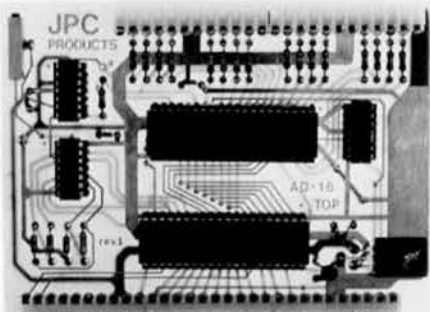
Thanks to your acceptance of the **SCREDITOR**, and in a bit of silliness because of the bulk of our first son a couple of weeks ago, we are now offering this fantastic program at the **NEW LOW PRICE** of only

**\$69.95**

Manual only: \$14.95 Complete source listing: \$39.95

## JPC PRODUCTS FOR

# 6800 COMPUTERS



USES  
ONE  
I/O  
SLOT

### 16 CHANNEL A/D BOARD

- 8 BIT DATA
- SOFTWARE CONTROLLED GAIN
- 3300 SAMPLES PER SECOND
- $\pm 0.7\%$  ACCURACY

**COMPLETE KIT: AD-16 \$69.95**

Terms: Cash, MC or Visa; Shipping & Handling \$3.00



Order Phone (505) 294-4623  
 P.O. Box 5815  
 Albuquerque, N.M. 87185

## Model EP-2A-79 EPROM Programmer



Software available for F-8, 6800, 8085, 8080, Z-80, 6502, 1802, 2650, 6809, 8086 based systems.

EPROM type is selected by a personality module which plugs into the front of the programmer. Power requirements are 115 VAC 50/60 Hz. at 15 watts. It is supplied with a 36-inch ribbon cable for connecting to microcomputer. Requires 1 1/2 I/O ports. Priced at \$155 with one set of software. (Additional software on disk and cassette for various systems.) Personality modules are shown below.

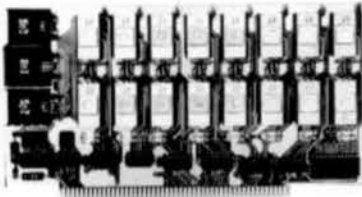
Part No.	Programs	Price
PM-0	TMS 2708	\$15.00
PM-1	2704, 2708	15.00
PM-2	2732	30.00
PM-3	TMS 2716	15.00
PM-4	TMS 2532	30.00
PM-5	TMS 2516, 2716, 2758	15.00
PM-X	MCM68764	33.00

**Optimal Technology, Inc.**  
 Blue Wood 127, Earlyville, Virginia 22936  
 Phone (804) 973-5482

# DIGITAL RESEARCH COMPUTERS

(214) 494-1505

## 16K EPROM CARD-S 100 BUSS



**\$59.95**  
KIT

FIRST TIME OFFERED!  
BLANK PC BOARD - \$28

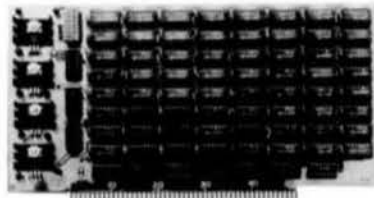
USES 2708's!

Thousands of personal and business systems around the world use this board with complete satisfaction. Puts 16K of software on line at **ALL TIMES!** Kit features a top quality soldermasked and silk-screened PC board and first run parts and sockets. Any number of EPROM locations may be disabled to avoid any memory conflicts. Fully buffered and has WAIT STATE capabilities.

OUR 450 NS 2708'S  
ARE \$8.95 EA. WITH  
PURCHASE OF KIT

ASSEMBLED  
AND FULLY TESTED  
ADD \$30

## 8K LOW POWER RAM KIT-S 100 BUSS SALE



PRICE  
CUT!

**\$119.50**  
KIT

21L02  
(450 NS RAMS!)

Thousands of computer systems rely on this rugged, work horse, RAM board. Designed for error-free, NO HASSLE, systems use.

Blank PC Board w/Documentation  
\$29.95

Low Profile Socket Set...\$13.50

Support IC's (TTL & Regulators)  
\$9.75

Bypass CAP's (Disc & Tantalums)  
\$4.50

ASSEMBLED AND FULLY  
BURNED IN ADD \$35

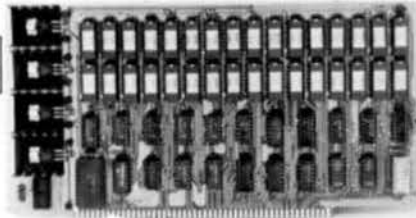
ALL ASSEMBLED BOARDS  
ARE TESTED AT 4MHZ.

## 16K STATIC RAM KIT-S 100 BUSS

PRICE CUT!

**\$225**  
KIT

FOR 4MHZ  
ADD \$10



### KIT FEATURES:

1. Addressable as four separate 4K Blocks
2. ON BOARD BANK SELECT circuitry (Crommed Standard). Allows up to 512K on line!
3. Uses 2114 (450NS) 4K Static Rams
4. ON BOARD SELECTABLE WAIT STATES
5. Double sided PC Board, with solder mask and silk screened layout. Gold plated contact fingers
6. All address and data lines fully buffered
7. Kit includes ALL parts and sockets
8. PHANTOM is jumpered to PIN 67
9. LOW POWER: under 1/8 amps TYPICAL from the +5 Volt Buss
10. Blank PC Board can be populated as any multiple of 4K.

BLANK PC BOARD W/DATA-\$33

LOW PROFILE SOCKET SET-\$12

SUPPORT IC'S & CAPS-\$19.95

ASSEMBLED & TESTED-ADD \$35

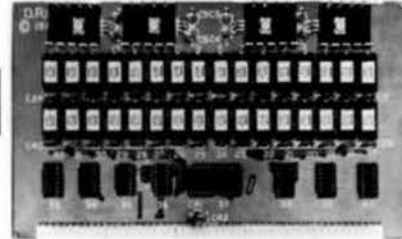
OUR #1 SELLING  
RAM BOARD!

## 16K STATIC RAM SS-50 BUSS

PRICE CUT!

**\$229**  
KIT

FULLY STATIC  
AT DYNAMIC  
PRICES



FOR SWTPC  
6800 BUSS!

ASSEMBLED AND  
TESTED - \$35

### KIT FEATURES:

1. Addressable on 16K Boundaries
2. Uses 2114 Static Ram
3. Runs at Full Speed
4. Double sided PC Board. Solder mask and silk screened layout. Gold fingers
5. All Parts and Sockets included
6. Low Power: Under 1.5 Ampe Typical

BLANK PC BOARD-\$28

COMPLETE SOCKET SET-\$12

SUPPORT IC'S AND CAPS-\$19.95

## NEW! STEREO! S-100 SOUND COMPUTER BOARD NEW!

At last, an S-100 Board that unleashes the full power of two unbelievable General Instruments AY3-8910 NIMOS computer sound IC's. Allows you under total computer control to generate an infinite number of special sound effects for games or any other program. Boards can be called in BASIC, ASSEMBLY LANGUAGE, etc.

### KIT FEATURES:

- TWO G.I. SOUND COMPUTER IC'S
- FOUR PARALLEL I/O PORTS ON BOARD
- US 3 ON BOARD AUDIO AMPS OR YOUR STEREO
- ON BOARD PLOT TYPING AREA
- ALL SOCKETS, PARTS AND HARDWARE ARE 1 CLU ED
- PC BOARD IS SOLDERMASKED, SILK SCREENED, WITH GOLD CONTACTS
- EASY, QUICK, AND FUN TO BUILD, WITH FULL INSTRUCTIONS
- US 3 PROGRAMMED I/O FOR MAXIMUM SYSTEM FLEXIBILITY

Both Basic and Assembly Language Programming examples are included

### SOFTWARE:

SCL Interpreter coming soon! Our new Sound Command Language Interpreter along with the Register Examine/Modify (REM) routines and Sound Effects Library (SEL) will be available soon in EPROM. SCL makes sound effects programming generally easier and quicker than that written in Basic or Assembly Language. An SCL users group will be formed, and the best new SCL programs submitted will be added to the Sound Effects Library in EPROM.

COMPLETE KIT!  
**\$84.95**

(WITH DATA MANUAL)

BLANK PC  
BOARD W/DATA  
\$31

Perfect for  
OEM's

**\$169.95**

WIRED!  
NOT A KIT!

**4 MHZ**

## S-100 Z80 CPU CARD



ASSEMBLED AND TESTED! READY TO USE! Over 3 years of design efforts were required to produce a TRUE S-100 Z80 CPU at a genuinely bargain price!

### BRAND NEW!

- 7 or 4 MHZ Operation
- Generates MWRITE, so no front panel required
- Jump on reset capability
- 8080 Signals emulated for S-100 compatibility
- Top Quality PCB. Silk Screened. Solder Masked. Gold Plated Contact Fingers

### NEW! G.I. COMPUTER SOUND CHIP

AY3-8910. As featured in July, 1979 BYTE! A fantastically powerful Sound & Music Generator. Perfect for use with any 8 Bit Microprocessor. Contains: 3 Tone Channels, Noise Generator, 3 Channels of Amplitude Control, 18 Bit Envelope Period Control, 2-8 Bit Parallel I/O, 3 D to A Converters, plus much more! All in one 40 Pin DIP. Super easy to interface to the S-100 or other busses.

SPECIAL OFFER: \$14.95 each Add \$3 for 60 page Data Manual.

FOR SALE!  
4MHZ  
LOW POWER - 300NS  
2114 RAM SALE!  
4K STATIC RAMS. MAJOR BRAND, NEW PARTS.  
These are the most sought after 2114's, LOW POWER and 300NS FAST.  
8 FOR \$44

**Digital Research Computers**  
(OF TEXAS)

P.O. BOX 401565 • GARLAND, TEXAS 75040 • (214) 494-1505

TERMS: Add \$1.00 postage, we pay balance. Orders under \$15 add 75¢ handling. No C.O.D. We accept Visa and MasterCard. Tex. Res. add 5% Tax. Foreign orders (except Canada) add 20% P & H. 90 Day Money Back Guarantee on all items. Orders over \$50. add 85¢ for insurance.



# Series 2000 Brings it all Together!



#### Hardware Features

- 2 MHz. 80800 MPU
- Double Floppy Disk Drive • 368K bytes formatted
- 32K, 48K, or 64K byte dynamic RAM
- Intelligent Video Terminal
- Commercial typewriter keyboard with function keys and numeric pads
- 2 RS-232C serial ports

#### Software Features

- UCSD Pascal™ System Software Package
- 6800 Multi-tasking System (MTS6800)
- Business BASIC Compiler
- WORDMATE™ Word Processor
- Various Application Packages

™ UCSD Pascal is a trademark of the Regents of the University of California.

#### Packaging

- Attractive, Compact, desk-top enclosure
- Light-weight, highly portable
- Provision for 3 I/O Expansion modules
- Highly reliable, ease of maintenance

**Price:** • Quantity 1 (one) end user price **\$2,995** • Attractive OEM/Dealer Discounts Available



**WAVE MATE INC.**  
18005 Adria Maru Lane  
Carson, California 90746  
213-532-4532  
Telex 194369

**EUROPEAN HEADQUARTERS  
WAVE MATE INTERNATIONAL**  
159 Ch de Vleurget  
1050 Bruxelles, Belgium  
(02) 649-1070 Telex 24050



## 6809!

### INTRODUCING THE NEW STATE-OF-THE-ART IN MICROCOMPUTER SOFTWARE

#### OS9-1 SINGLE USER

##### OS9-1 WITH TAPE FILE MANAGER

	on 2716's	\$ 95.00
	on 2708's	\$ 85.00
Manual & Source	only	\$ 95.00

##### OS9-1 WITH DISK FILE MANAGER

	on 2716's	\$170.00
	on 2708's	\$150.00
Manual & Source	only	\$150.00

##### DEBUGGER PACKAGE

(approx 1K)

	on 2716's	\$ 55.00
	on 2708's	\$ 35.00
	on tape	\$ 35.00
	on disk	\$ 35.00
Manual & Source	only	\$ 35.00

##### INTERACTIVE EDITOR/ASSEMBLER

	on 2716's	\$150.00
	on 2708's	\$120.00
	on tape	\$ 75.00
	on disk	\$ 75.00
Manual & Source	only	\$ 75.00

Above items available after approx. June 1, 1980.

See GIMIX Ad  
Pages 3 & 4



### COMING SOON!!! BASIC09 OS9-2 MULTIUSER

When ordering, you must specify: type of CPU card, type of disk controller, size of media and starting address for your I/O ports.

From the company that puts it all together. GIMIX, SMOKE, SWTPC, MICROWARE, ANADIX, SPINWRITER, DIGITUS, HI-PLOT, MICROWORKS. . .

## H H H ENTERPRISES

BOX 493, Laurel, MD.  
ZIP 20810  
PHONE 301-953-1155

## ED SMITH'S SOFTWARE WORKS 6809 SOFTWARE TOOLS

**RRMAC M6809 RELOCATABLE RECURSIVE MACROASSEMBLER.** The one assembler that contains real macro capabilities (see our NOV/DEC AD). RRMAC is designed with the assembly language programmer in mind and contains many programmer convenience features. RRMAC contains a mini-editor, supports spooling or co-resident assembly, allows insert files, is relocatable, generates cross-references, execution times, lists target addresses of all relative references.

**M69RR . . . . . \$150.00**

**SGEN M6809 DISASSEMBLER/SOURCE GENERATOR** will produce source code (with symbolic labels) suitable for immediate re-assembly or re-editing. The output source file can be put on tape or disk. A full assembly type output listing with labels and mnemonic instructions can be printed or displayed on your terminal. Large object programs can be segmented into small source files.

**M89RS . . . . . \$50.00**

**ANNOUNCING TWO NEW M6809 CROSS DEVELOPMENT TOOLS CROSSBAK**—A 6809 TO 6800 CROSS MACROASSEMBLER that runs on your M6809 development system to produce relocatable M6800 object code. Has all features of M69RR (see above). Includes 6800 Linking Loader.

**M69CX . . . . . \$200.00**

**CROSSGEN**—A 6800 OBJECT CODE DISASSEMBLER/SOURCE GENERATOR that runs on your M6809 development system. Has all features of M89RS (see above). An invaluable tool for converting all those 6800 object files over to the M6809.

**M69CS . . . . . \$75.00**

All programs are relocatable and come complete with Linking Loader, Programmer's Guide and extensively commented assembly listing. Available on 300 Baud cassette or mini-floppy disk. For disk, specify SSB or FLEX. Source Text input/output is TSC/SSB editor/assembler compatible.

Dealer inquiries welcome.

FLEX is trademark of TSC

### Ed Smith's SOFTWARE WORKS

P.O. Box 339, Redondo Beach, CA 90277, (213) 373-3360

## CT-64

## CT-1024

- ★ **FAST! Average Screen Writing Speed: 19K baud**
- ★ Memory Mapped Video Adapter for your CT-64 or CT-1024.
- ★ Just plug it in and go.
- ★ The terminal works like normal until the supplied output routine is used, then the CT-64 or CT-1024 display works at processor speed.
- ★ The J.B.I. Video Board takes 1 main SS50 slot.
- ★ Video Memory can be dip switch selected to any 1K memory slot.
- ★ Combine the J.B.I. Video Board with our up and coming pseudo graphics adapter board and you put new life into the old CT-64.
- ★ The J.B.I. Video Board comes built and is jumper selectable for either the CT-64 or CT-1024.
- ★ If your terminal is CT-1024 we need to know if it is a standard 32 characters per line; or has been modified for 64 characters per line.
- ★ **SHIPPING NOW!**

The J.B.I. Video Board sells for:  
\$169.00 with your 21LO2'S  
\$179.00 with our 21LO2'S  
(8-21LO2'S are required)

We have been a dealer for SWTPC since 1976.

**Johnson Micro Computer**  
2607 E. Charleston  
Las Vegas, Nevada 89104  
1-702-384-3354

Mastercharge and Visa accepted  
Dealer inquiries invited

# Software Source Books<sup>TM</sup>

Combining detailed descriptions with complete source listings, these books explain the internal operations and algorithms used in Hemenway Associate's popular systems software.

How much would such a complete software resource cost? If you've seen the PAPERBYTE books by Jack Hemenway and Robert Grappel you know how inexpensive they can be. And now you can have the companion volumes to the RA6800ML macro assembler and LINK68 linking loader books.

Remember, these are not just books; they are SoftwareSourceBooks ---- complete Software resources! Order them today; VISA and MasterCard accepted.

## <sup>TM</sup> CP/68 - OPERATING SYSTEM (\$34.95)

- \* PIP Peripheral Interchange
- \* Program transfers data between physical devices
- \* Wildcard Filenames and Extensions
- \* Relocatable anywhere in Memory
- \* Extended Instruction set includes 6809-type instructions (PSHX, PULX, etc)
- \* Device-independent I/O
- \* Random and Sequential Files
- \* Fits in less than 8K
- \* Chaining and overlaying
- \* Single Supervisor Call furnishes all DOS services
- \* Easily interfaced to new devices and peripherals
- \* Dynamic file allocation

## <sup>TM</sup> STRUctured Basic Language (STRUBAL+ ) COMPILER for both business and scientific uses (\$49.95)

- \* Variable precision from 4 to 14 digits
- \* Structured Programming forms
- \* Produces Relocatable and linkable code
- \* COMMON and DUMMY sections
- \* Extensibility
- \* String Handling
- \* Full scientific package
- \* Data structures with mixed data types

## XA6809 Macro Linking Cross Assembler (\$24.95)

- \* Runs on any M6800
- \* Full Macro facilities
- \* COMMON section for the production of ROMable code
- \* Conditional Assembly
- \* Generates linkable and relocatable code
- \* Sorted Symbol table listing
- \* Hash-coded Symbol table for speed

=====

Hemenway Associates Inc. 101 Tremont St. Boston MA 02108

Name	Title	Company
Street	City	State Zip

( ) Check enclosed in the amount of \$.....

( ) Bill VISA                      ( ) Bill MasterCard

Card No..... Exp. Date.....

Please send the following books:

-----

Add \$0.75 per book to cover postage and handling

# 6809!

## INTRODUCING THE NEW STATE-OF-THE-ART IN MICROCOMPUTER SOFTWARE.

COMING SOON!!!

OS-9 ~ The most advanced microcomputer operating system available. Similar to UNIX™ (Bell Telephone Labs) with enhancements for the 6809 hardware environment. A two level firmware system. Level one supports BASIC09 and all I/O, memory management and MPU time. Level two supports all the above and inter-process communications as well as a hierarchical directory system. The addition of level two (can be done at a later date) allows full timesharing, multiprogramming real time operation for those systems supporting extended memory mapping and an interrupt-driven disk system. A minimum of 4K RAM required for level one.

A new BASIC09 language system embracing the best of BASIC, PASCAL and other popular high level languages running under OS-9 with speed and utility unmatched by other BASIC or PASCAL versions. An excellent choice for business, scientific and/or personal computer users. We feel this is the most advanced BASIC available for any microcomputer.

*Call or write today for our  
free catalog.*

## MICROWARE

5835 Grand Ave. • P.O. Box 4865  
Des Moines, IA 50304 • 515/279-8844

## '68' MICRO JOURNAL

- ★ The only ALL 6800 Computer Magazine.
- ★ More 6800 material than all the others combined:

### MAGAZINE COMPARISON

(2 years)

Monthly Averages  
6800 Articles

KB	BYTE	CC	DOBB'S	TOTAL PAGES
7.8	6.4	2.7	2.2	19.1 ea. mo.

Average cost for all four each month: \$5.88  
(Based on advertised 1-year subscription price)

'68' cost per month: \$1.21

That's Right! Much, Much More

for About

1/5 the Cost!

1-Year \$14.50   2 Years \$26.00   3 Years \$36.50

-----  
OK, PLEASE ENTER MY SUBSCRIPTION

Bill My: Master Charge ☐ — VISA ☐

Card # \_\_\_\_\_ Exp. Date \_\_\_\_\_

For ☐ 1-Year   ☐ 2 Years   ☐ 3 Years

Enclosed: \$ \_\_\_\_\_

Name \_\_\_\_\_

Street \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_

My Computer Is: \_\_\_\_\_

### 68 MICRO JOURNAL

3018 Hamill Road  
HIXSON, TN 37343

Foreign surface add \$9.50 per year.  
Foreign Air Mail add \$29.00 per year.



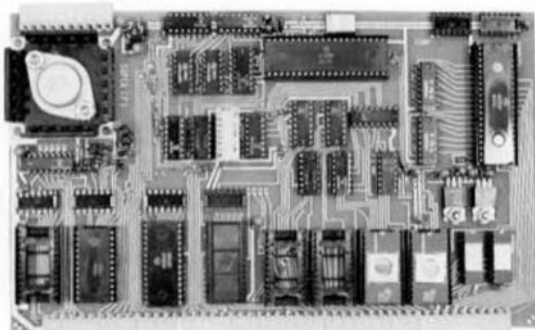
Life subscription \$175.00

NOTE: CANADA & MEXICO ADD \$4.50 per year surface.  
New subscriptions require 6-8 weeks processing.

## NOW INDUSTRIAL QUALITY AT LOW COST

### FROM THOMAS INSTRUMENTATION

Industrial system boards are now available separately for OEM, prototyping or hobbyist applications. Shipped from stock, these are the same quality cards used in monitors and machine tool controls designed for GM, LTI, and General Electric. All cards are SS-50 buss compatible and are suitable for dedicated applications. The CPU card and the Video RAM Card may be combined on a TI backplane as a stand-alone micro — ideal for prototypes or hobbyists. ATTENTION OEM's: If you have a control data acquisition, monitoring, or other microprocessor application. Check with TI for more information about custom software design for the TI CPU or any other 6800 series system. TI also has non-SS-50 buss single board 6800 systems.



### SS-50 SUPER CPU

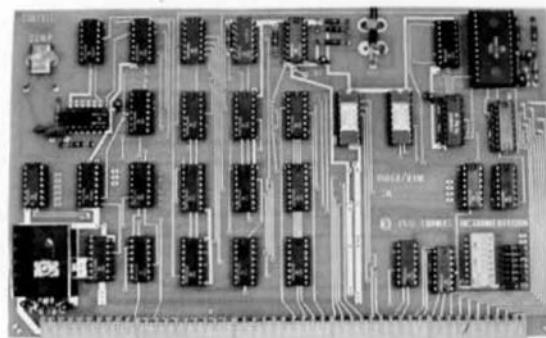
- SS-50 or stand alone computer
- 1K of RAM at \$A000 I/O on board at \$A400 (Relocatable)
- 2K Monitor (Mikbug compatible) in 2708 EPROM
- 2 8 bit parallel ports with 2 control bits and power
- RS-232 ACIA port, 2nd TTL ACIA optional
- 3 16 bit counter/timers (expandable to 6 add 2nd 6840)
- 128 byte RAM at 0000 is jumper selectable
- Battery back-up for 32 bytes of RAM
- Plug back to back with Video RAM for Stand alone micro or customized smart terminal

**ASSEMBLED \$195.00**  
**CARD AND DOCUMENTATION \$49.00**

### SS-50 VIDEO RAM

- Fully synchronous operation — No jitter
- 7 by 9 Characters Programmable reverse video
- Full 128 Character ASCII set
- 1K of memory can be mapped to any 1K boundary
- Full documentation includes software (Replaces OUTEE)

**ASSEMBLED \$149.00**  
**CARD, CRYSTAL and DOCUMENTATION \$39.00**



TI SS-50 Wire-Wrap Card .....	24.00
TI SS-50 Parallel I/O Card .....	95.00
Card only .....	35.00
GIMIX 16K Static RAM w/Soft addressing .....	368.00
GIMIX 16 RAM without Soft addressing .....	298.00
3, 4, 7 SLOT Backplanes (per slot) .....	4.00

TI cards available from stock

## THOMAS INSTRUMENTATION

168-8th Street, Avalon, N.J. 08202  
Phone (609) 967-4280



## DEALERS FOR GIMIX SWTPC SSB

**CALL FOR DEALER, OEM. AND QUANTITY PRICES**



# SMOKE SIGNAL BROADCASTING

Presents

## 3 Powerful New SS-50/SS-50C Boards

### **DCB-4** **Disk Master** Double Density Controller Board and DOS68D Double Density DOS **\$449.00**

The new DCB-4 is a truly state-of-the-art development which allows up to 366K bytes to be stored on a single 5¼" disk and has these outstanding features:

- Up to four 5¼" and four 8" drives can be handled in the same system with a user definable logical unit table. (DOS68D will be compatible with future hard disk systems).
- Under software control, the user can select the following for any drive:
  - ☆ Single sided or double sided operation.
  - ☆ Single density or double density data.
  - ☆ 5¼" or 8".
  - ☆ Stepping Rate.
  - ☆ 40 track or 35 track density on double sided 5¼" drives.
  - ☆ User can select the system boot configuration.
- Occupies only 16 bytes of memory space (F760-F76F standard). User selectable to any 16 byte address space.
- Can read and write a single sector by itself. On-board buffer memory allows full interrupt capability in interrupt driven systems. Once data transfer has been initiated, no more processor time is required.
- Contains extended decoding circuitry for extended addressing per SS-50C bus which can be enabled by an option jumper.
- SSB provides a means for copying software written by older versions of DOS68 to be read by DOS68D. All new media formatted by DOS68D can be read by all older versions of DOS68. DOS68 is SSB's 6800 disk operating system.
- Track 0 of side 0 is recorded in single density per IBM standard.
- Phase-locked-loop assures highest data integrity attainable.

All of these features are available for immediate delivery on one standard 5½" x 9" 50 pin SS-50/SS-50C card for only \$449.00. The price includes DOS68D version 5.1, MONITOR object code on diskette, and a manual with the source listing.

### **SCB-69** **Super Computer Board** 6809CPU Board **\$299.00**

The most versatile 6809 CPU Board on the market is now available from Smoke Signal Broadcasting and has the following features:

- Standard 2 MHz operation. (Shipping 1.5 MHz until August 80)
- 20 bit address generation for up to 1 Mbyte of memory. Uses an improved address translation RAM which is compatible with present extended addressing schemes yet requires much less overhead when used in multi-user systems.
- All on-board devices can be switch selected to occupy any or all extended pages. Any on-board device may be disabled and its memory space is then available for external memory.
- Standard real-time clock (time-of-day, day-of-week, day-of-month) with battery back up capable of generating programmable interrupts.
- Up to 20K of EPROM can be installed on the CPU Board.
- Standard 1K of RAM on board.
- Includes improved 6809 Monitor (and source listing).
- Contains an FPLA for decoding EPROM address and optional devices. Switches are used to select 2K/4K EPROM and Fast/Slow I/O.
- Contains provision for optional 9511/9512 floating point processor.
- NMI line is user selectable to work with either SS-50 or SS-50C busses.

Price for the new SCB-69 is only \$299.00 for an assembled, burned-in fully tested board.

### **M-32-X** **32K** **Memory Board** **\$539.00 \$439.00**

The first and only 32K Static Ram Board on standard size (5½" x 9") SS-50/SS-50C Bus Circuit Card is made by Smoke Signal.

- Switch selectable to any 4K boundary.
- Any 4K block may be switch enabled or disabled.
- Fully compatible with SS-50C extended addressing (allows memory decoding up to 1 Mbyte).
- Extended addressing capability may be switched off for compatibility with SS-50 systems.
- Gold Bus Connectors for high reliability.
- Guaranteed 2MHz operation (tested at 2.2 MHz).
- Low power consumption — 8 volts at 2.4 amps typical.

M-32-X 32K Memory Board is priced at \$539.00.

M-24-X 24K Memory Board expandable to 32K, is \$439.00.

And our M-16-X 16K board is back to the old price of \$299.00.

# SMOKE SIGNAL



# BROADCASTING®

31336 Via Colinas, Westlake Village, CA 91361, (213) 889-9340

# THE SOURCE

FOR

# UCSD PASCAL<sup>TM</sup>

ON THE  
6809

IMMEDIATE DELIVERY FOR SWTPC USERS, 8" or 5¼" DISKETTE  
SMOKE SIGNAL BROADCASTING USERS, INQUIRE

CSI-1 Operating Systems, PASCAL Compiler, Screen Editor, Filer,  
Linker, Library, Setup, Binder, Interpreter, BIOS .....\$250.00

CSI-2 BASIC Compiler, YALOE, (Line-editor for hard-copy terminals),  
Patch, Disassembler, Calculator.....\$100.00

CSI-3 MACRO Assemblers for 6809, 6800 and other Microprocessors ..\$100.00

ALL THREE DISKS and MANUAL (SYSTEM) .....\$419.00

**FREE! UCSD PASCAL USER'S MANUAL PLUS SWTPC  
IMPLEMENTATION NOTES WITH PURCHASE OF CSI-1**

68000 Coming, Summer, 1980  
68000 Assembler, Spring, 1980

**OEM and DEALER INQUIRIES INVITED**



1317 CENTRAL AVE. KANSAS CITY, KANSAS 66102

CALL TOLL-FREE (800) 255-4411

Continental U.S.A. Only  
(Kansas Residents call 913/371-6136)

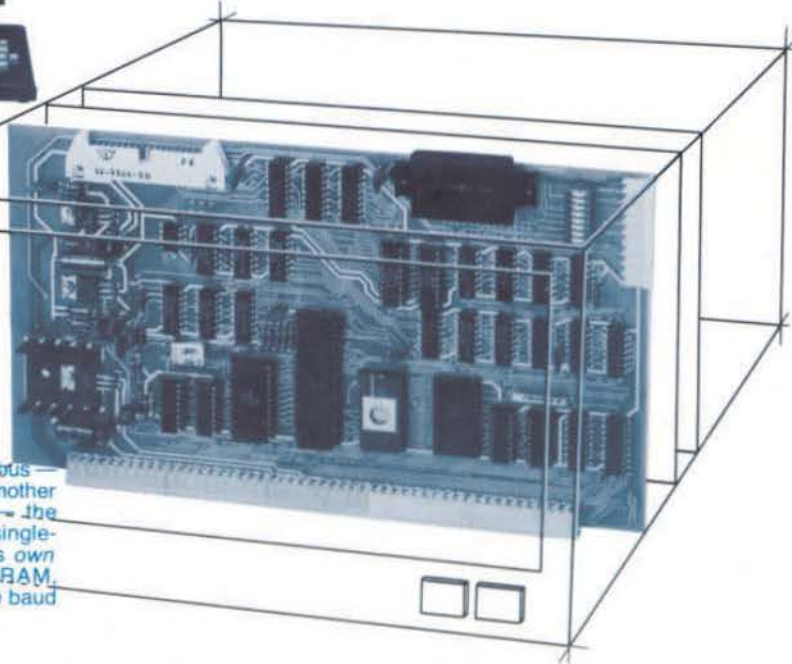


"UCSD Pascal" is a registered trademark of The Regents of The University of California



## 6809 PROCESSING POWER! The Percom SBC/9. Only \$199.95.

Fully compatible with the SS-50 bus requiring no modification of the mother board, memory or I/O slots — the SBC/9™ is also a complete, single-board control computer with its own ROM operating system, RAM, peripheral ports and a full-range baud clock generator.



**Make the SBC/9 the heart of your computer and put to work the most outstanding microprocessor available, the 6809.**

### the Mighty 6809

Featuring more addressing modes than any other eight-bit processor, position-independent coding, special 16-bit instructions, efficient argument-passing calls, autoincrement/autodecrement and more, it's no wonder the 6809 has been called the "programmers dream machine."

Moreover, with the 6809 you get a microprocessor whose programs typically use only one-half to two-thirds as much RAM space as required for 6800 systems, and run faster besides.

And to complement the extraordinary 6809, the Percom design team has developed PSYMON™, an extraordinary 6809 operating system for the SBC/9™.

### PSYMON™ — Percom System MONitor

Although PSYMON™ includes a full complement of operating system commands and 15 externally callable

™ trademark of Percom Data Company, Inc.

utilities, what really sets PSYMON™ apart is its easy hardware adaptability and command extensibility.

For hardware interfacing, you merely use simple, specific device driver routines that reference a table of parameters called a Device Control Block (DCB). Using this technique, interfacing routines are independent of the operating system.

The basic PSYMON™ command repertoire may be readily enhanced or modified. When PSYMON™ first receives system control, it initializes its RAM area, configures its console and then 'looks ahead' for an optional second ROM which you install in a socket provided on the SBC/9™ card. This ROM contains your own routines that may alter PSYMON™ pointers and either subtly or radically modify the PSYMON™ command set. If a second ROM is not installed, control returns immediately to PSYMON™.

- Provision for multi-address, 8-bit bidirectional parallel I/O data lines for interfacing to devices such as an encoded keyboard.
- A serial interface Reader Control output for a cassette, tape punch/reader or similar device.
- An intelligent data bus: multi-level data bus decoding that allows multiprocessing and bus multiplexing of other bus masters.
- Extended address line capability — accommodating up to 16 megabytes of memory — that does not disable the on-board baud rate clock or require additional hardware in I/O slots.
- On-board devices which are fully decoded so that off-card devices may use adjoining memory space.
- Fully buffered address, control and data lines.

The SBC/9™, complete with PSYMON™ in ROM, 1K of RAM and a comprehensive users manual™ costs just \$199.95.

# PERCOM

PERCOM DATA COMPANY, INC.  
211 N. KIRBY GARLAND, TEXAS 75042  
(214) 272-3421

Percom 'peripherals for personal computing'

To place an order or request additional literature call toll-free 1-800-527-1592. For technical information call (214) 272-3421. Orders may be paid by check, money order, CDD or charged to a VISA or Master Charge account. Texas residents must add 5% sales tax.

PRICES AND SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE